THE PI GETS AI
First look at the official AI accelerator

MOD UBUNTU
Enhance your install with secret settings

PODCAST IT!
The apps, mics and tips for a pro podcast

LINUX FORMAT

The #1 open source mag

SECURE YOUR CHAT
The message services that really protect your texts

BOOST YOUR VPN PRIVACY

Tweak and hack your network settings for snoop-free browsing and downloading

GET A FREE VPN! p32

PLUS: HOW TO

- Code classic Amiga-style demos
- Edit and control PDFs without Adobe
- Create a modern note tool in Fyne

DIY PI ROUTER
Build your own custom Pi-based Wi-Fi router

RAY TRACING
Enhance your Steam Deck gaming graphics

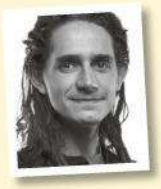
MS OFFICE RESCUE
Save old Office docs that Microsoft can't

LXF Summer 2024
FUTURE

LINUX FORMAT

MEET THE TEAM

We're exploring how VPNs work to help boost privacy, but what open source web resource would we find that you'd just visited if we were spying on you?



Jonni Bidwell
The **LXF** archives have all the answers, both computational and existential. Self-promotion aside, the DigitalOcean tutorials are impeccably written. **Gamingonlinux.com** deserves a mention, too – for keeping us up to date on which games broke due to invasive copy protection.

Michael Reed
OK, you knew I was a nerd... This year, I'm improving my data science skills with Python. I have a rare disorder, and I am studying data from my own body, and I'd like to have a go at hosting datasets. I'm trying to decide between Google's Kaggle or CERN's Zenodo.

Mayank Sharma
You'll catch me on Mastodon. Compared to Twitter's noise and spam, it is a breath of fresh air. Its threads have meaningful discussions, something that's evaporated from Twitter. Its philosophy and design encourages purposeful exchanges among diverse participants, unlike Twitter's echo chambers.

Les Pounder
I find myself going to **Archive.org** many times each week. Why? For access to the plethora of programming books necessary to learn older machines. From Basic to Forth, I've dabbled with them all, thanks to this awesome and necessary resource.

Matthew Holder
In a mythical world where VPN security was non-existent, a spy would see me regularly checking in on *Home Assistant,* which runs on a Raspberry Pi 4, viewing security cameras hosted with *Frigate,* or they'd see the audio books I am listening to, using Audiobookshelf.

*Savings are based on the cover price.

Twitchy curtains

A UK politician once said, "People don't need curtains, but they do seem to seem to like having them and awful lot." And another UK politician once said, "Why should I have my right to look into someone's home impinged by a curtain?" I'm not sure if this says more about the state of UK political debate or our obsession with window decorations. The point is that it does offer an insight into how politicians view people's privacy, as in it's something the state has control over and an interest in controlling.

As we point out in our in-depth feature *(on page 32),* the burgeoning $45 billion VPN industry should highlight just how popular keeping your online tracks hidden is, and who can blame curtain-loving people? With acronym agencies all over the world keeping tabs on the general public, corporations tracking us all at every opportunity, governments constantly trying to weaken encryption, logging everything done online, nevermind running hacking teams, we're not surprised.

But how do you close your digital living room curtains so you can keep your digital life private? We're going into just how VPNs work – and they all use open source OpenVPN or WireGuard – how you can configure them, create your own and run the free-tier trusted Proton VPN, alongside giving you a few recommendations and how to choose a better VPN. With the UK government logging all the websites you browse, hopefully this can make your online life a little more enjoyable!

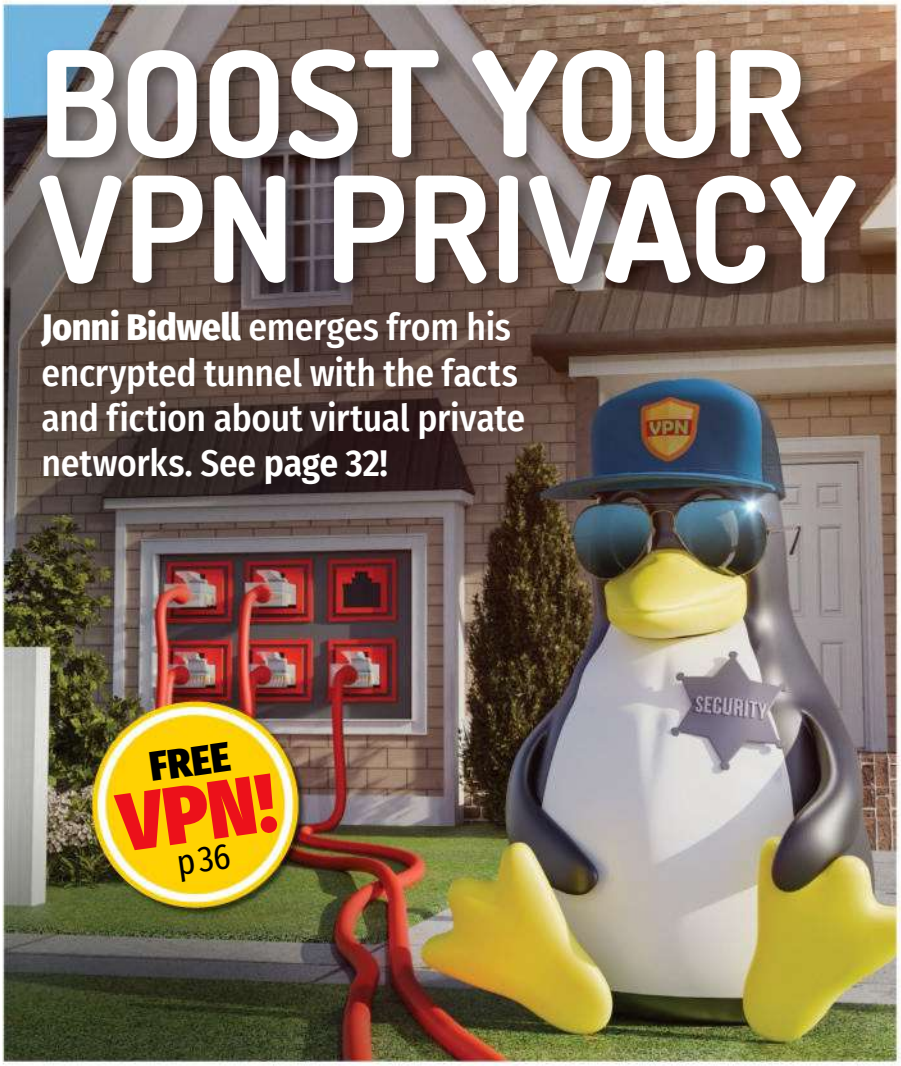
Neil

Neil Mohr Editor
neil.mohr@futurenet.com

Contents

SUBSCRIBE NOW!
Page 16

REVIEWS



CREDIT: Magictorch

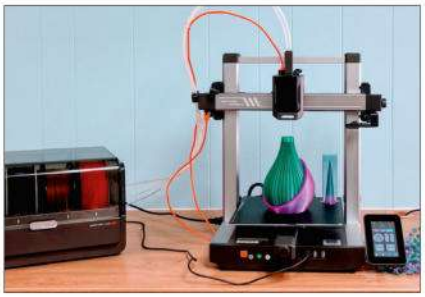
ROUNDUP

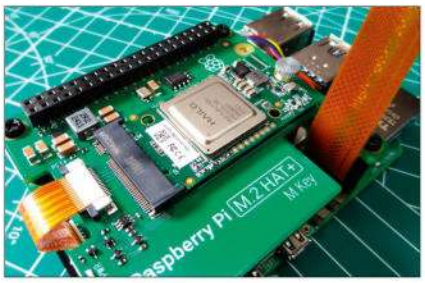
IN DEPTH

CREDIT: Getty Images, Magictorch

CONTENTS

Pi USER





CODING ACADEMY

REGULARS AT A GLANCE

TUTORIALS

CREDIT: CD Projekt

ADMINISTERIA

Newsdesk

THIS ISSUE: EU backtracks on chat » RISC-V laptop looks likely » Fedora gets AI trim » MINIX maker marked by award

PRIVACY

EU backs down on chat control

EU Council has withdrawn the vote on its controversial Chat Control plan, which would have severely undermined end-to-end encryption.

At the start of 2022, the EU Commission proposed the Regulation to Prevent and Combat Child Sexual Abuse. Its stated aims were to prevent the distribution of child sexual abuse material (CSAM) online. But the proposed methods caused controversy.

Meredith Whittaker, president of secure messaging service Signal, said as much in a statement released on 17th June entitled "New branding, same scanning."

Whittaker pointed out that in November 2023, the EU Parliament voted to exclude end-to-end encryption from mass surveillance. However, the demand for client-side scanning arose under the name "upload moderation".

The EU narrative seems to run along the lines that end-to-end encryption isn't compromised by this kind of monitoring, as files and messages are scanned prior to being encrypted.

Whittaker called out this kind of "rhetorical games", pointing out that any monitoring of this kind "fundamentally undermines encryption".

Posting on eX-Twitter, Edward Snowden also stated, "EU apparatchiks are trying to legislate a terrible mass surveillance measure, despite universal public opposition (no sane person wants this), by inventing a new word for it: upload moderation."

It would seem that the EU agreed to some extent. In April, leaked documents showed that EU interior ministers requested an amendment to the regulation. This change would have exempted professional accounts of staff of intelligence agencies, police and military from the scanning of chats and messages. This seems to be a tacit admission that this kind of mass scanning would risk breaching those users' privacy. When covering the leak, outlets such as the EU Reporter argued that said privacy should also be enjoyed by private individuals and businesses.

There's also controversy over whether the regulation in its current form is even feasible. In theory, the messaging app would generate a hash for files that are shared, which could be compared to hashes for known CSAM.

As Whittaker points out, the fact that this kind of scanning is done automatically client-side rather than remotely or via a back door bypasses end-to-end encryption.

When similar legislation was debated in the UK, services like WhatsApp and Signal threatened to pull out of the country rather than comply. However, even if they kowtowed, there'd be nothing to stop users installing messaging software that doesn't deploy upload moderation. In light of this on 21st June, the EU Council decided to withdraw the vote on the Chat Control plan proposed by Belgium, which included provisions for upload moderation. At the time of writing, the draft law has been postponed indefinitely.

Signal president Meredith Whittaker has made it clear that the platform will not cooperate with mass surveillance laws of this kind.

CREDIT: X, Alamy, Framework

HARDWARE

Framework RISC-V model

Framework and DeepComputing are working on a mainboard with a JH7110 processor.

Images of the new mainboard have yet to be released but it will be compatible with the Framework 13 laptop.

On 18th June, Framework announced plans to introduce a DeepComputing RISC-V mainboard powered by a StarFive JH7110 processor.

The main blog article notes that this is a huge step forward for Framework, as well as an opportunity to make RISC-V more accessible. This is ironic, given that at the time of writing, the mainboard is out of stock, but interested parties can opt for an email notification when it becomes available for purchase.

The JH7110 processor has four U74 RISC-V cores from SiFive, a company that uses the RISC-V ISA. StarFive integrates said CPU cores with other peripherals, while DeepComputing created the mainboard capable of leveraging the processor. Framework, in turn, is making the laptops that use this mainboard. While this may sound rather anarchic, the article points out that such is the benefit of an open ecosystem.

Frameworks advises, however, that this mainboard is intended mainly for "developers, tinkerers and hobbyists to start testing and creating on RISC-V". It frankly admits that the peripheral set performance lags far behind its Intel and AMD-powered mainboards.

The board also isn't especially customisable. Due to the limitations of the processor, it has soldered memory and uses microSD cards/eMMC for storage.

On the plus side, DeepComputing has been working with the development teams at both Canonical and Fedora to ensure the mainboard works with their respective versions of Linux, though initially support will be experimental.

Framework has advised that the best way to keep abreast of developments is via its online marketplace. Find out more: **https://frame.work/products/deep-computing-risc-v-mainboard**.

DISTRO

Fedora 41 boosts AI development

AI edition includes features requested by community.

While Christian Schaller is Red Hat's director for desktop, graphics and AI acceleration, he wears many hats, given he is also deeply involved with GStreamer, Gnome and Fedora.

This is why in mid-June, Schaller posted an update on the Gnome Blogs about Fedora Workstation's Artificial Intelligence edition.

As incredulous as some commenters were, there's no doubting that the latest version of Fedora will include features that have long been requested by the community.

In the blog post, Schaller states that Red Hat has been working with IBM to release Granite AI models that follow a largely transparent framework. He also points out that Red Hat plans to release the full source code for the LLM, as well as launch the Instructlab project. This will ostensibly make it easier for developers to download a Granite LLM model and train it to their specific needs.

He also underscored the work being done on Fedora to ensure it's compatible with other AI projects, such as Llama LLM. He singles out Red Hat dev Tom Rix for special mention, who has been working on bringing in AMD-accelerated support (ROCm) for PyTorch to Fedora.

As an optimised tensor library for deep learning using GPUs and CPUs, the stated aim is to have PyTorch work on Fedora with hardware accelerated using any major chipset.

Other new features include better Nvidia GPU compatibility and better HDR support. Read the post at **https://tinyurl.com/fedoraai**.

OPINION

SUMMER RELEASE!

Italo Vignoli
is one of the founders of LibreOffice and the Document Foundation.

"*LibreOffice 24.8* will be announced in the second half of August, and the developers are working hard to optimise the new features that will be included.

It will be the first major release with a significant number of contributions in the area of accessibility and RTL (right-to-left – Arabic and a large number of Asian languages), thanks to the work of Michael Weghorn and Jonathan Clarke.

In addition, there will be several new features in each of the four main modules:

Writer: improvements to document and character formatting, hyphenation and table management.

Calc: addition of several new functions, and improved interoperability of pivot tables with *Microsoft Office's* proprietary OOXML format.

Impress: improved note management in Normal view, improvements to current templates, and faster opening of *MS Office's* proprietary PPTX files.

Charts: new Pie-of-Pie and Bar-of-Pie chart types, which make it possible to import the same charts created in proprietary OOXML format.

Math: new features for a module used in education.

Plus numerous additions in the area of import and export filters, and password-based encryption functionality for standard ODF files."

OPINION

DRAGGED DOWN

David Stokes
is a technology evangelist at Percona.

Support for *MySQL 5.7* ended in October, so users have had months to carry out testing, choose where to move, and conduct migrations. However, the volume of *MySQL* server instances out there has hardly dropped.

We are all told that being outside support is a risk. So why aren't people shifting their workloads? Why are they sticking with 5.7?

One of my colleagues started testing different versions of *MySQL* to check performance using *Sysbench* and *TPC-C*, using the default settings each database provided. They found that v5.7 was significantly better than the default versions of 8.036 and 8.4. So, why move with such a noticeable performance impact?

The impact is significant. For those who love *MySQL*, seeing performance drop in new versions is concerning for the health of this open source project. While Oracle is investing heavily in its own paid services, it is not bringing those performance improvements nor other enhancements to the open source version. This is a significant risk for *MySQL*. The community is ready to contribute and keep *MySQL* healthy, if we have the opportunity to do so. Without improvement, users will stick with *MySQL 5.7* and the project could face a slow death.

OPERATING SYSTEM

ACM celebrates MINIX

Association for Computing Machinery has awarded Andrew S Tanenbaum the Software System Award for creating the first version of MINIX.

The **ACM**'s Software System **A**ward is given to innovators who solve real-world problems, like **P**rofessor **T**anenbaum, who developed M**INIX**.

CREDIT: Jantangring, CC BY-SA 4.0, https://en.wikipedia.org/wiki/Andrew_S._Tanenbaum

In 1987, Professor Andrew S Tanenbaum created the Unix-like operating system MINIX at Vrije Universiteit in Amsterdam. It was designed as a companion to his textbook *Operating Systems: Design and Implementation.* To this end, the source code was made freely available and a truncated version of it is printed in the book.

Linux users all owe him a great debt of gratitude for this, given that Linus Torvalds himself was a MINIX user. The early Linux kernel was also written on a MINIX host system and adapted various features such as the OS's filesystem.

Although Linux has long since stopped using MINIX code, on 18th June the ACM awarded Tanenbaum for his key contributions. The official announcement also praises MINIX for being the basis for the MeikOS operating system on Meiko transputer-based computers.

Later iterations of MINIX 3.0 are designed for resource-limited/embedded computers

Read more: **www.acm.org/media-center/2024/june/technical-awards-2023**.

SOFTWARE

Mozilla rolls out trusted AI

Firefox nightly build now includes some trustworthy AI features.

Mozilla's official blog has announced the inclusion of AI features in the latest nightly build.

This seems to be the culmination of a number of efforts, given that in late May an experimental AI accessibility feature was introduced to *Firefox 130* to generate alt-text for PDF images. Full support for web browsing is also apparently in the pipeline.

The blog post stresses that all AI features in the nightly build will be on an opt-in basis only. The initial line-up includes ChatGPT, Google Gemini, HuggingChat and Le Chat Mistral.

Read more at: **https://tinyurl.com/firefoxai**.

Firefox's new **AI** features will only be included in the nightly build and will be on an opt-in basis.

DISTRO

Ubuntu 24.10 Nvidia updates

Ubuntu will soon see Nvidia defaulting to Wayland over X11.

Traditionally, Ubuntu has defaulted to X.org sessions in Gnome due to the proprietary nature of the Nvidia graphics driver. Still, the upcoming Nvidia R555 driver has hugely improved Wayland support.

This is most likely why the desktop session in the upcoming 24.10 release of Ubuntu will use Wayland for Nvidia graphics.

To this end, in mid-June Ubuntu desktop developer Daniel van Vugt updated the GDM 46.0-2ubuntu2 session manager to remove the rules that make X.org the default for Nvidia.

Read more about the Ubuntu 24.10 roadmap at **https://tinyurl.com/ubuntu2410discourse**.

Ubuntu **D**iscourse lists other exciting developments in v24.10, including an overhauled installer and **A**pp **C**entre, and better Flutter integration.

CREDIT: Mozilla, Ubuntu

Distro watch

What's behind the free software sofa?

ENDEAVOUROS 2024.06.25

Five years ago, a handful of Antergos moderators decided to create a Linux distro of their own, despite having very little experience with OS development. The result was this rolling release, which is based on Arch Linux. EndeavourOS offers easy setup via the *Calamares* installer, which has been updated to 24.06.1.2. EndeavourOS's offline installer and live environment has also now been updated to deploy the Plasma 6.1 desktop. Read the full release notes at **https://endeavouros.com**.

EndeavourOS uses Plasma 6.1 and has an updated installer.

TAILS 6.4

Tails is based on Debian and is primarily designed to run in live mode for internet anonymity. Almost all connections are routed through Tor. The operating system now stores a random seed on the USB stick to secure the RNG. This is available both in live mode and for users who enable persistent storage. Tails also now uses HTTPS for Debian and Tails repositories, instead of **.onion** addresses, which supposedly makes the Additional Software feature more reliable. You can learn more at **https://tails.net.**

Tails now accesses HTTPS repos rather than the dark web.

EASYOS 6.0

EasyOS is officially an experimental distro. It uses much the same tech and package formats as Puppy Linux, which explains its logo. It deploys custom container technology named Easy Containers, each of which can run apps or an entire desktop environment. The latest release includes a huge number of built-in packages, including *Chromium, LibreOffice* and *Flowblade*. The OS is only available as a drive image (IMG) file, designed to be installed to USB. See **https://easyos.org**.

EasyOS allows apps to be run via multiple Easy Containers.

SUSE LINUX ENTERPRISE 15 SP6

SUSE Linux Enterprise (SLE) stands apart from the community-driven OpenSUSE in that it's specifically designed for business use. This means it's released less frequently but packages usually go through more extensive testing. The latest release is no exception, given SUSE claims it's "designed to meet the most stringent security and compliance standards". The Linux kernel has been updated to version 6.4. The OS also contains memory management upgrades as well as an improved filesystem. Learn more at **www.suse.com**.

SLE now has better security and network performance.

SDESK 2024.06.22

Like EndeavourOS, SDesk is based on Arch Linux. It also makes use of the *Calamares* installer for an easy setup. The current release comes with the latest versions of *Calamares* and *Octopi*. The *Swirl* browser has better accessibility features and supports localisation, too. SDesk also now uses the linux-zen kernel by default, resulting in multiple optimisations and scheduler improvements. The Blur My Shell Gnome extension has been removed. Read more at **www.stevestudios.net**.

The linux-zen kernel boosts performance and has more modules.

OPINION

STREAM LEARNING

Aaron Boxer
is a senior software developer at Collabora.

"One of GStreamer's key strengths as a multimedia framework is its support for a wide range of hardware and software platforms. GStreamer's machine learning support continues this tradition by integrating ONNX Runtime, a high-performance cross-platform inference engine for Open Neural Network Exchange (ONNX) models.

ONNX is an open standard format for representing machine learning models, allowing models to be trained on one framework and transferred to another for inference. ONNX Runtime runs multiple platforms including x86 and ARM CPUs, Nvidia, AMD and Intel GPUs, and AMD FPGAs.

GStreamer has supported ONNX for a number of years via its onnxobjectdetector element, which can detect objects in a video stream using a single-shot detector model, using CPU or CUDA.

To address a broader range of analytics workflows, the plugin was redesigned to make it simpler to support other AI models. It's now also possible to support other inference engines, such as TensorFlow Lite or PyTorch.

It is now easier than ever to create complex, multi-model GStreamer analytic pipelines that take full advantage of the latest state-of-the-art AI models running efficiently on the latest hardware."

OPINION

OUT OF TIME

Jon Masters is a kernel hacker who's been involved with Linux for over 22 years, and works on energy-efficient Arm servers.

"I happened upon a post to the Linux Kernel Mailing List titled "Proposition for fixing Y292B bug". At first, I thought, "That's a typo, it must be a reference to the (infamous) 2038 bug."

As it happens, it was not a typo. It was about when the 64-bit time_t Unix time will overflow in 292 billion years from now. I figured it must be a joke, but the author seemed seriously interested in discussing this distant 'problem'. Among the more gentle replies came one from Ted Ts'o, who noted the importance of trade-offs in computer science, and in particular that using more space to store time in a generic (infinitely scaling) manner would cost more than it was worth. After all, Ted noted, "Trying to make sure that software will work in the year 292 billion AD might not be something that most people would consider high priority. After all, it's … unlikely … that the x86_64 architecture will still be what we will be using 290 billion years from now."

I'm also not convinced that x86 will be widely used in 290 billion years. Which reminds me, my Qualcomm X Elite-based Dell XPS has arrived and I'm going to be playing with getting Linux up and running on it. I'll write about my experience once I get it working."

Kernel Watch

Jon Masters summarises the latest happenings in the Linux kernel, so that you don't have to.

Linus Torvalds announced the final set of Linux 6.10 Release Candidate kernels as the development cycle for 6.10 came to a close and anticipation was building for the opening of the 6.11 merge window (the period of about two weeks at the start of a development cycle during which disruptive patches are allowed). We will have a full summary of all the new features coming in 6.11 in next month's *Kernel Watch*.

RIP Daniel Bristot de Oliveria

Many of us were saddened to hear of the sudden death of Daniel Bristot de Oliveria at the age of just 37. Daniel was well known in the Linux real-time and scheduling communities, in particular for his work applying formal methods to the analysis of Linux kernel behaviour. Formal methods apply mathematically rigorous techniques to analyse and verify that the behaviour of a system matches the specification. In English, this means proving that a system does what it is supposed to, something that is very important, in particular, for those using systems like Linux in safety-critical applications, such as automotive (cars).

> "I was blown away by **Daniel's ability to solve seemingly intractable problems (such as functional safety for Linux kernels) and make it seem like it was an 'easy' topic.**"

Daniel had a PhD in applying particular formal techniques to the analysis and verification of the Linux kernel. He created tools such as the upstream RV (Runtime Verification), which was being applied to the formal analysis of the kernel for use in safety-critical environments.

He also authored *RTLA (Realtime Latency Analysis* tool), which provides analysis of real-time performance and debugging based on models generated from RV. This marked a revolutionary approach to kernel debugging that relied on BPF and tracers to build live models of the current system state, as opposed to the historical method of black-box statically-generated system states.

He wrote about this work back in 2021 in Red Hat's *Research Quarterly* publication. I briefly worked with Daniel but did not know him well. I was, however, blown away by his ability to solve seemingly intractable problems (such as functional safety for Linux kernels) and make it seem as though it was an 'easy' topic in casual conversation.

Various members of the Linux scheduling and real-time communities have written about their shock at the news of Daniel's passing. In particular, Linux Weekly News had a guest piece written by a number of his friends. Perhaps if there's one thing we can take away from this it is to always live life to its fullest, because you just never quite know what might happen. LXF

» ONGOING DEVELOPMENT

Kirill A Shutemov posted a patch that makes five-level paging support unconditional for x86-64. This means that CONFIG_X86_5LEVEL will go away. Linux uses page tables to cooperatively manage memory between itself and underlying hardware that must translate virtual addresses as seen by application software into physical locations in RAM. This facilitates the abstraction of many different processes isolated from one another. As systems become bigger, the tables describing memory grow in depth, with five levels being required for 56-bit physical addressing on x86 Now, five levels won't require any special handling.

There is some discussion of the merits of paravirt scheduling, aka breaking the abstraction of virtualisation slightly to allow cooperation between a host and its guest VM instances to understand "where the VCPU is physically running". The aim is to address issues seen with "latencies, power consumption, resource utilisation etc". Apparently, various off-list discussions are ongoing.

Answers

Got a burning question about open source or the kernel? Whatever your level, email it to **answers@linuxformat.com**

Neil Bothwick undergoes nightly builds to ensure he's up to date.

Q Not so shortcuts

I use *Chromium's* search engine shortcuts and have many set up to make searching specific sites quick and easy. Now I have a second laptop and would like to set up the same set of shortcuts on there, but there appears to be no way to export them. I have tried grepping the files in **.config/chromium** to see which contain my search strings but am having no luck. Is there a relatively easy way of transferring the settings from one computer to another, or am I doomed to having to retype everything?
David Pearce

A *Chromium* stores settings, history and most other information in an *SQLite* database in **~/.config/chromium/Default/Web Data**. You can manipulate this with the *SQLite* command-line client but there's a couple of graphical options that make the job easier. One is *DB Browser for SQLite* (**https://sqlitebrowser.org**), but we used *SQLite-web* (**https://github.com/coleifer/sqlite-web**) for this. This is a Python program, so if it is not in your distribution's repositories, you can install it with:

```
$ pip install sqlite-web
```

Now open the **Web Data** file from the old computer with:

```
$ sqlite-web "Web Data"
```

This tries to open a web page in your default browser. If this is *Chromium*, you will get an error, because *Chromium* will lock its database file. Either set your default browser to something else, or close *Chromium* as soon as it opens and then open **http://127.0.0.1:8080** in another browser. Select the keywords table, press the Export button, pick JSON as the export filetype and *SQLite-web* will save a file to your **Downloads** folder. Exit *SQLite-web* by pressing Ctrl+C in the terminal running it.

Now run it again with **Web Data** from the new computer. Select the keywords table and press Drop to delete it. Then create a new table called keywords (we do this because there is no option to empty the table without deleting it). Select the keywords table and press Import, pick the file you just exported and then you can exit *SQLite-web*. Your changes are saved automatically. Finally, restart *Chromium* and your search engine shortcuts should all be there.

Q Wiping media

I think that it would be very useful to empty the **/media/user** folder at startup because if an old mounted device folder is still there, it may break some applications that use it. In that case – for example, if the system had a hard reset – this folder (example: **NT2**) is not deleted and when restarting the system, the same storage device is mounted on a **NT21** folder and the application (any) will not recognise the **NT2** folder and be broken.
Lucas Flynn

A We can think of three ways to deal with this. The first is brutal: delete everything in **/media** at boot time. Create the file **/etc/rc.local**, as root, containing:

```
#!/bin/sh
rm -fr /media/*
```

Then run:

```
$ sudo chmod +x /etc/rc.local
```

This creates an executable script that is run when the system boots, deleting everything in **/media**. Another option is to have udisks, the service that handles mounting of removable drives, mount the drives in **/run/media/$USER** (the default in some distros) by creating the file **/etc/udev/rules.d/90-udisks2.rules** containing:

```
ENV{ID_FS_USAGE}=="filesystem|crypto",
ENV{UDISKS_FILESYSTEM_
SHARED}="0"
```

Change the **0** to a **1** to mount under **/media**. As **/run** is a tmpfs filesystem, it is automatically cleared upon rebooting.

If you like the idea of using a tmpfs but need to keep the current locations to avoid breaking other things, you could make **/media** a tmpfs, too. Add this line to **/etc/fstab**:

```
tmpfs  /media  tmpfs  defaults  00
```

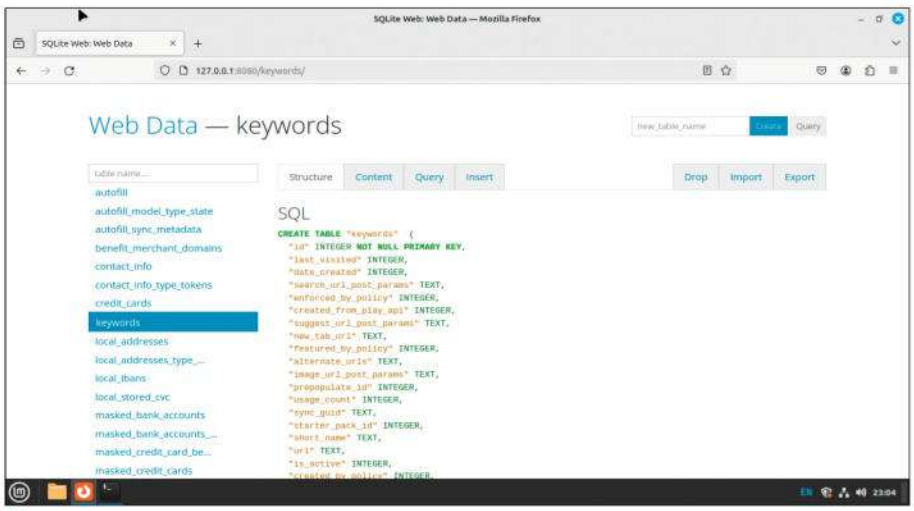
Make sure you have nothing in **/media**, then either reboot or run:

```
$ sudo mount /media
```

This creates an in-memory filesystem at **/media**, so any mount points for removable devices are also created in memory and disappear upon rebooting.

Q Video capture

I have access to some training videos through a web portal, but would like to download them to view offline when travelling. The portal does not have an option to download and I cannot use *youtube-dl* or *yt-dpl as* I have to go through a login page. Is there a way to record the videos to a file for later



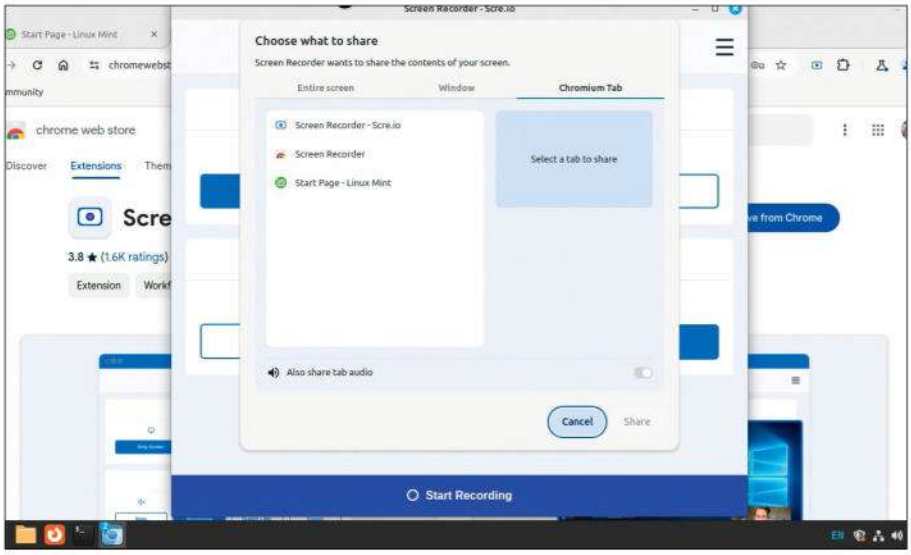
Chromium stores much of its data and settings in an SQL database, so manipulating it is not a trivial task.

viewing? My preferred browser is *Chrome* if that helps, but any solution is welcome.
Oliver Lynch

A This is possible, but first consider why the portal does not allow downloading. Would you be infringing any licensing by downloading them to your computer? If you are happy to do this, there is a *Chrome* extension called Screen Recorder; search for it in the Chrome Web Store. This extension not only works within the browser, it can also record the entire desktop or an application window, but in your case the option to record a browser tab is what you want. Once installed, find the video you want to record, then run the extension, which opens in a separate window. Tell it you want to record a browser tab, click the box to record audio if you need that, too, then select the browser tab from the list it gives you. Now start the video. The extension records in real time – there is no option to download the video as fast as your internet connection allows, so you may have to leave it to complete. Once it is complete, stop the recording and press the Save button.

Here you may encounter the only real limitation we have with Screen Recorder: it only records in WebM format. This is handled by some but not all video players, so you may prefer to transcode it into a more universal format, such as MP4.

For this task, we recommend *HandBrake* (**https://handbrake.fr**), a graphical video conversion program that enables you to select the part of the video that you wish to transcode, useful both for removing unwanted portions of the video and for selecting a short segment for an initial test of your settings. *HandBrake* provides many profiles for different conversion formats and playback devices, so you can select the ideal combination of size, container format, quality and conversion speed. It also enables you to queue multiple conversions for batch processing.



Chrome's Screen Recorder can record video you play in the browser, as well as create screencasts of your desktop.

Q How long is a string?

I have the problem that I have a very long command and it's too long for me to run in one line. The command is basically installing packages. I have tried to do this:

```
apt install \
pkg1 \
pkg2 \
pkg3 \
...
```

However, it gives:

```
E: Unsupported file / given on commandline
```

How do I install packages in a multilined fashion?
Ewan Stephenson

A You are not limited by the maximum number of characters your terminal can display on a single line. It is quite acceptable to type a command that flows on to a second or subsequent line of the terminal window. Until you press Enter, it is all treated as one line. Using backslashes to continue a line is a convenience – it makes the command more readable – but the shell combines all the lines into a single command before executing it. So, whichever way you do it, the OS limit for the maximum command and argument length applies. The good news is that this limit is usually huge, far more than you would ever type on a single like – you can see what it is with this command:

```
$ getconf ARG_MAX
```

On our system, that returns 2,097,152 – over two million characters. Good luck typing that, with or without backslashes!

This means that your issue is not caused by your splitting of the package list but by something else. It looks from the error message that you are including a forward slash somewhere in the command. Also, you should not have a backslash at the end of the final line.

If you are ever generating a command list that is too long, you can use *xargs* to

» A QUICK REFERENCE TO... RCLONE

There are so many choices for cloud storage, and each one has its own driver or utility program for Linux, or none at all, leaving you with only a web interface. Wouldn't it be great to have one program that worked with all (or most) cloud providers, running on Linux and providing a consistent experience whatever the provider? There is: it is called *Rclone* (**https://rclone.org**) and it supports around 70 cloud storage facilities. *Rclone* is a command-line program that works in a similar way to *Rsync*, so the learning curve should be quite shallow if you have already used the latter. After installing, the first step is to set up a connection to your cloud provider by running:

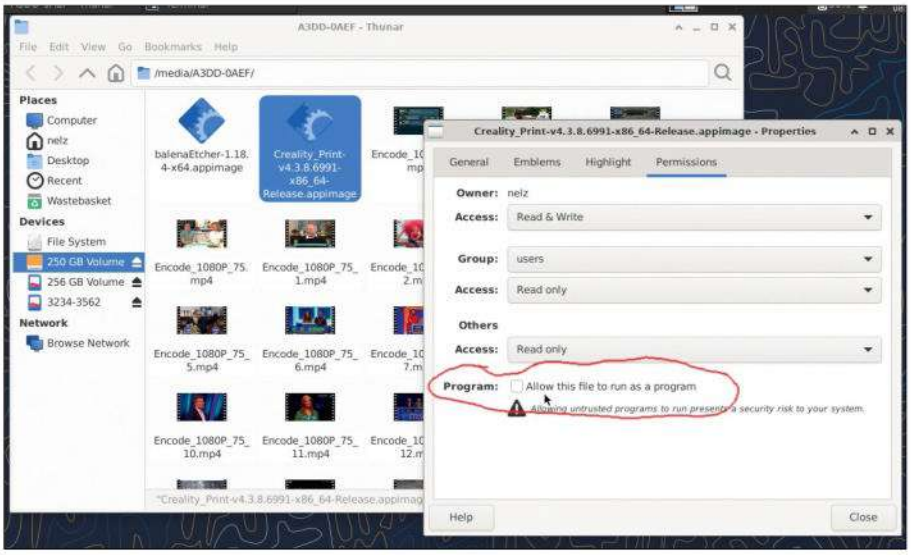
```
$ rclone config
```

This lists any existing connections and has options to edit or remove them, or add a new connection. When picking that option, you are walked through the steps needed to set up that provider. Once done, usage is similar to *Rsync*, for example:

```
$ rclone copy ~/Documents
cloud1:Documents
```

This copies the contents of your **Documents** folder to a similarly named folder on the given cloud provider. As with *Rsync*, existing files are not copied, only new or altered files. Unlike *Rsync*, files missing from the source are not deleted from the destination. If that is what you want, use the *Sync* command instead.

There are several other options, all well documented, so give it a try.



Windows filesystems have no concept of Linux permissions; reformat your USB stick to run programs from it.

process it. This command is most often used to pass the output of one command to another – for example, to process a list of files generated by *find*. If you run something like the following, *xargs* installs as many packages at a time as is allowed by system limitations:

```
$ echo "pkg1 pkg2 ..." | xargs apt install
```

Within these limitations, you can also set the maximum number of arguments with **-n** or the maximum line length in characters with **-s**. You can also create input from a file, so to install five packages at a time from a list in a file, you would use:

```
$ xargs -n 5 apt install <list.txt
```

Note that options for *xargs* go before the command to be executed. Any options given later are passed through to the command itself.

Q Sticky problem

I run Linux Mint 21.1 Vera on Xfce. I would like to run AppImages from a USB stick, similar to portable apps in Windows. Is this possible? When I try, I am unable to tick the box Allow The File To Run As A Program. It all works fine on my hard drive. I suspect it is probably a permissions issue, but in reality I have no real clue.
Bailey Richards

A It is a permissions issue, because your USB stick is formatted with a Windows filesystem. When you put your AppImage file on a Linux filesystem, you are able to set the executable permission. That is what the Allow The file To Run As A Program box does. When you do this on a Windows filesystem, that status cannot be saved, so the file is never executable.

There's a couple of ways to work around this that are not to be advised as they tend to circumvent the security provided by udisks. If you want to be able to run programs directly from a USB stick, the most reliable approach is to format your stick with a Linux filesystem. You could use ext4, but some recommend using ext2 as the ext4 journal can potentially increase wear on the flash drive. If you also need to be able to read the drive on Windows, you could create two partitions on it, the first using VFAT for Windows, so it can be automatically mounted on Windows. Then put a Linux filesystem on a second partition that contains your AppImage files. It is possible to install a driver on Windows to read an ext2 filesystem, but this should not be necessary in your case as the AppImage files are only for use in Linux.

Q Build demolition

I want to delete build directories from everywhere. Can I do this without manually going inside all directories?
Zak Davidson

A What you want to do is easy to achieve, but potentially very dangerous. Before you attempt this, you should ask yourself exactly what you are trying to achieve, and whether this the best way to go about it. You may have a number of **build** directories left over after compiling software from source – those should be safe to delete. You may also have **build** directories that were created by your package manager when installing software. It is not a good idea to mess with those unless you know exactly what you are doing.

You can find the directories easily enough with the *find* command:

```
$ find / -type d -name build
```

This shows all directories below the root filesystem with the given name. You can use shell wildcards with **-name**, but then you should enclose the string in quotes or the shell interprets the wildcards before calling *find:*

```
$ find / -type d -name 'build*'
```

If you have other filesystems mounted that you do not wish to search, trawling through a large network share really slows things down, so you can use the **-xdev** option to instruct *find* to not cross filesystem boundaries:

```
$ find / -xdev -type d -name build
```

Once you have your list of directories, and have checked carefully that it contains nothing that could cause breakage, you can delete them by using one of these two commands:

```
$ find / -type d -name build -delete
$ find / -type d -name build -exec rm -fr {} +
```

The first simply tries to delete each directory – however, this only works on empty directories, which may or may not suit your objective. The second uses the **-exec** option to pass the list to a shell command, in this case `rm -fr` , with the braces replaced with the list of found files. That deletes the directories and their contents. You may get some errors about directories not being found, because they have already been deleted. So, if you have, say, **build/build**, *find* first finds **build** and then **build/build**, and then can't find the latter to delete it as it has already gone. The **-ignore_readdir_race** option tells *find* to ignore these errors

These operations are not reversible, so be very sure that you really want to proceed. If possible, take a backup or snapshot before deleting anything. LXF

GET HELP NOW!

We'd love to try to answer any questions you send to answers@linuxformat.com, no matter what the level. We've all been stuck before, so don't be shy. However, we're only human (although many suspect Neil is a robot), so it's important that you include as much information as you can. If something works on one distro but not another, tell us. If you get an error message, please tell us the exact message and precisely what you did to invoke it.

If you have, or suspect, a hardware problem, let us know about the hardware. Consider installing *hardinfo* or *lshw*. These programs list the hardware on your machine, so send us their output. If you're unwilling, or unable, to install these, run the following commands in a root terminal and send us the system.txt file, too:

```
uname -a > system.txt
lspci >> system.txt
lspci -vv >> system.txt
```

Mailserver

WRITE TO US

Do you have a burning Linux-related issue that you want to discuss? Write to us at *Linux Format*, Future Publishing, Quay House, The Ambury, Bath, BA1 1UA or email **letters@linuxformat.com**.

No Etchings

This issue keeps happening every time I try to transfer the Pop!_OS ISO into my USB – whenever I click Flash From File, it stops for a minute or two and just displays an error opening the source. I've done almost everything – running the app as admin, changing files, and so on. The error code is '(0 ,h.requestMetadata) is not a function' – can you tell me what's going on to get this working?

Adam Edge

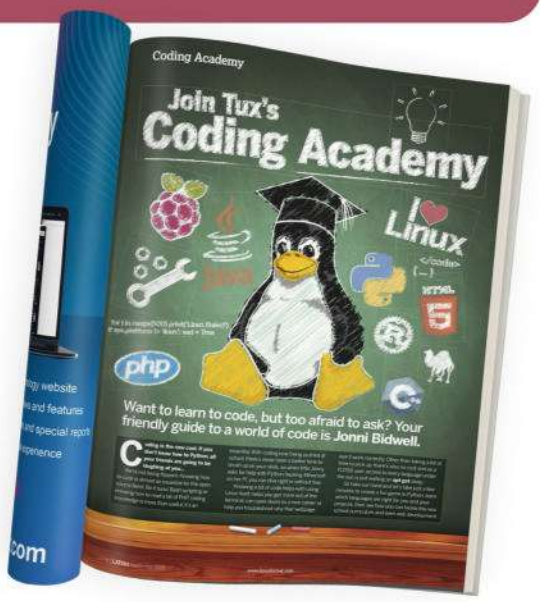
Neil says...

There's been reports of issues with certain builds of *Etcher* in the past – was this the stable build off the front page? I can see there's been reported issues for 1.19.x releases, so it's certainly not you. There's always the legacy 32-bit build that's version 1.7.9 – it's not going to be any slower...

It's worth eyeing these things up on their GitHubs. That's one of the cool things about open source: it's developed in the open, so you can see what's going on. If you're interested, head to **https://github.com/balena-io/etcher**, where there's full install info and you can look up past versions and current issues.

But you don't have to be stuck – there's plenty of other options. Another easy guided choice is **www.ventoy.net**, though that could be overkill. There's Fedora's image writer on Flathub (**https://flathub.org/apps/org.fedoraproject.MediaWriter**), or you could learn to use the universal *dd*, though that can be nerve-wracking for some (**https://bit.ly/lxf318dd**).

There's more than one way to skin a cat – is something people don't say any more.



We have run several features on getting you started coding, so it's not like we never cover the subject.

Can we code it?

What's the point in your coding section? I never know where to start with any of the projects. What would be good is a basic guide to starting coding. I've got an idea for a project but just don't know where to start, rather than these projects that seem so complicated.

Graham Collier

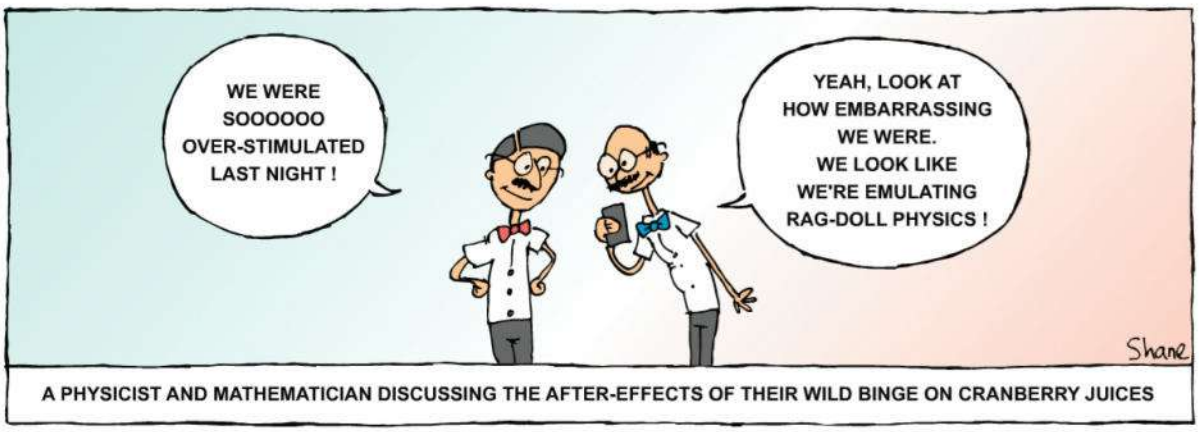
Neil says...

We're stuck between a rock and a hard place here. If we offer elementary coding basics, I think most people could pick this stuff up online for free and in a level of detail way beyond anything we could offer. We always advertise Code Club (**https://codeclub.org**), which has a host of starter projects for a range of core languages to help you get up and running.

There's also Code Academy (**www.codecademy.com**), and we like the challenges at **https://edabit.com/challenges**. There are free courses, too, such as those at **www.coursera.org/courses?query=free**, and, of course, there are books! Books from the library, online resources, Humble Bundle and beyond.

The projects we run are definitely above this level, but are generally aimed at giving you a mostly finished project that you can poke around, learn how it works and tinker with to expand your knowledge. Though

Helpdex



We wouldn't recommend Fedora for beginners, as it's not aimed at them.

Ferenc's current classic demo series might be pushing this a bit far with his crazy maths...

I expect we will run one of our popular Get Coding features in the future, but these tend to point you to the tools you need to get started, such as dealing with a nice GitLab for storing code, integrated development environments for an easier coding experience, and perhaps a starter project to get you up and running. We're less concerned with programming principles, like data types and loops. Let us know what you think.

Wearing Fedoras

When I was researching Linux before installing it, pretty much every distro for beginners was Debian-based, such as Mint, Pop!_OS, Zorin, Elementary or Ubuntu. If it wasn't one of those, it'd be Arch-based, such as Manjaro or Endeavour.

It was my curiosity that ended me up at Fedora Workstation, as no one really mentioned it. I found it was super-friendly for someone who barely even understood what a package manager was. So, how come people don't recommend Fedora for beginners? And what about distros like OpenSUSE, which I've heard is pretty similar to Fedora.

Tom Wyatt

Neil says...

Fedora is backed by Red Hat, one of the biggest Linux developers in the world, and now owned by IBM. Red Hat uses Fedora to test out cutting-edge technologies that it's planning to implement down the line on its super-stable server distros. Hopefully, that should point to why it's not recommended for beginners. It's not that it's isn't a good distro, it's just that its constantly under development and needs to be updated and maintained by the user.

There's lots of other reasons: as it's cutting-edge, the documentation isn't always there, and there's less support in forums or a library of past questions with solid answers – something you get with Ubuntu or Debian. It's also not aimed at anyone who wants to game or use the desktop for productivity, as Pop!_OS is.

None of this is to say Fedora is bad or poor – we're hoping to cover its next main release later this year, so look out for that! LXF

» LETTER OF THE MONTH

Not so intelligent AI

Having read a bit on AI, it seems that all the projects rely on Nvidia hardware – isn't there anything available for AMD? What about the Radeon AI? I thought I'd seen that was heading to Linux – are you going to be covering how that works? I'm hoping I won't be needing a Copilot key any time soon, but what about a Tux AI key?

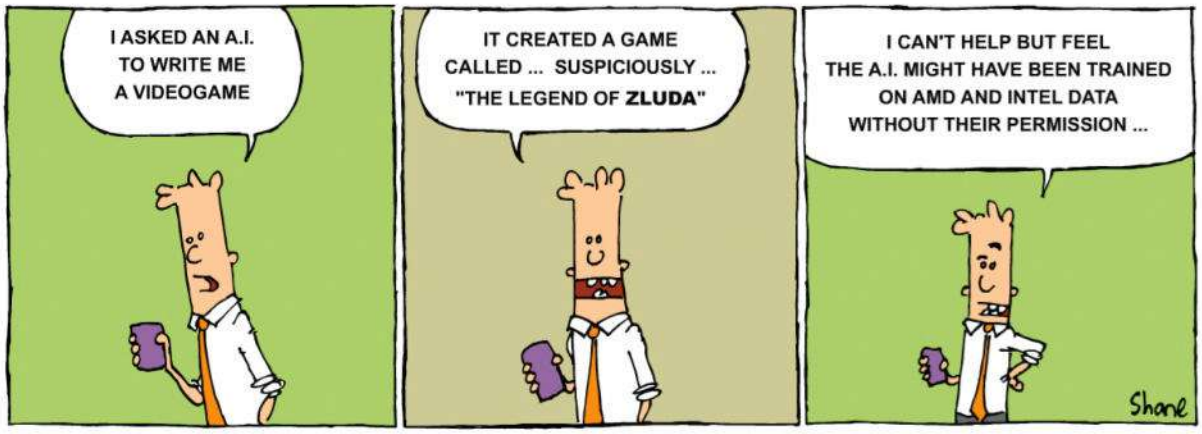
Troy Cooper

Neil says...

The reason all AI projects are focused on Nvidia hardware is that – to its credit – it largely cornered the GPGPU market back in the day with CUDA. Nvidia poured a ton of money into the best software support possible and it's paid off. So much so, it's now the most valuable corporation in the world, passing Apple and Microsoft just recently in its market cap, because every major and minor IT company is clamouring to buy Nvidia kit to power unproven AI services to run on that hardware.

It's not like AMD (and let's not forget Intel) doesn't produce decent GPU hardware, it's just that AMD's software has always been lacking. Unfortunately, OpenCL has always been 'still catching up' with CUDA on performance. But now Intel has finally decided to try to compete for compute pounds, so perhaps the two can do something to catch up. Both had been sponsoring a project called ZLUDA to translate CUDA into AMD's ROCm/HIP stack, but that doesn't appear to have really gone anywhere (possibly as it's on legally dubious ground, but we're not lawyers), so they're back to getting OpenCL to run more efficiently.

There's no denying the GPU powerhouse that Nvidia has become – and for good reason.



shane_collinge@yahoo.com

TAKE YOUR LINUX KNOWLEDGE FURTHER

Enhance your knowledge with in-depth projects and guides

FUTURE

Ordering is easy. Go online at:

magazinesdirect.com

Or get it from selected supermarkets & newsagents

WD Gold 24TB

For once, **Shane Downing** regrets shouting, "Go big or go home!"

SPECS

Size: 24TB
Model: WD241KRYZ
£/TB: £26.83
Interface: SATA 6GB/s
Form: 3.5-inch
Tech: CMR
RPM: 7,200
Max: 298MB/s
Cache: 512MB
Power: 6.8W
Noise: 32dBA
Limit: 550TB per year
MTBF: 2.5M hours
Warranty: Five years

The WD Gold 24TB comes as the largest capacity hard drive currently available – other 24TB drives exist, but nothing larger can be purchased. It competes with the best hard drives primarily by offering more storage, but at a higher cost per GB. While WD's Gold line does share the flash-based OptiNAND technology with its Red Pro NAS line of hard disk rives, it has a unique feature to help it stand out: ArmorCache.

Who would want ArmorCache? Someone who needs data protection without losing performance. In an environment that can use write caching, the Gold can safely write data out on power loss. In a stricter environment where write caching is denied, the Gold can maintain performance by leveraging the benefits of its flash and justifying that price tag.

Differences from the 22TB model are minor. The 24TB model goes from 291MB/s to 298MB/s for the maximum sustained transfer rate. It's still using a 10-platter solution, with the side benefit that it's also more power-efficient, operating at 6.8W instead of 7.1W. Otherwise, the warranty is the same, comparable to its direct peers at five years in length, with a 550TB/year workload limit and a 2.5 million hours MTBF.

Flashy features

WD's OptiNAND technology is WD's way of keeping up in the capacity chase. Certain data can be written to flash memory instead of the HDD platters, and as this is metadata, performance is also improved due to the flash's superior random access performance. The flash is also non-volatile and allows the drive to be run with or without a write cache, while having power loss protection even with the volatile cache enabled, as data can be flushed in time. This capability is known as ArmorCache and only applies to the 22TB and 24TB Gold. It's the primary selling point in comparison to competing drives, though with journaling filesystems, the actual benefit is questionable.

In our real-world *DiskBench* test, the 24TB Gold comes into its own. It has very fast read and write rates, and, on the whole, a good copy transfer rate. As the platters have a higher areal density and hold more data than the 22TB version, it outpaces that drive as well. It takes the top spot in the 50GB write test, and comes in just a hair behind the Toshiba N300 18TB in the 6.5GB read test, while it falls a bit further behind in the 50GB copy test and places fifth.

Currently the Gold, Red Pro, IronWolf Pro, Exos and SkyHawk AI lines all offer a 24TB retail model. These all have similar specifications and should perform similarly as well. If you're looking to maximise your storage capacity, particularly for desktop use, you can't really go wrong with any of these. The SkyHawk AI targets visual applications, including surveillance, and the IronWolf Pro provides a three-year data recovery service. The Red Pro is great for NAS, and the Exos X24 is a good all-round choice.

There's no doubting the vast amount of storage, but also the price.

The WD Gold costs more than any of the others, and as such it should only be chosen if you specifically need the ArmorCache feature. If you don't need the absolute highest density possible, the 22TB Gold has a lower price per GB, but at that point you're probably better off picking up the Seagate Exos X20 20TB – that's currently the lowest price per GB for a 20TB or larger hard drive. LXF

VERDICT

DEVELOPER: WD
WEB: www.westerndigital.com
PRICE: £644

FEATURES	8/10	EASE OF USE	9/10
PERFORMANCE	7/10	VALUE	5/10

No surprises, providing satisfactory performance for mechanical storage. It's pricey but includes unique features thanks to the OptiNAND and ArmorCache technologies.

» Rating 6/10

Ultramarine Linux 40

Nate Drake finds out what puts the 'Ultra' in this user-friendly Fedora remix from our friends in the Land of Smiles.

IN BRIEF

Ultramarine is fairly heavy on system resources but the result is a clean, graphically rich interface. Setup is simple, with a huge number of extra packages relative to stock Fedora.

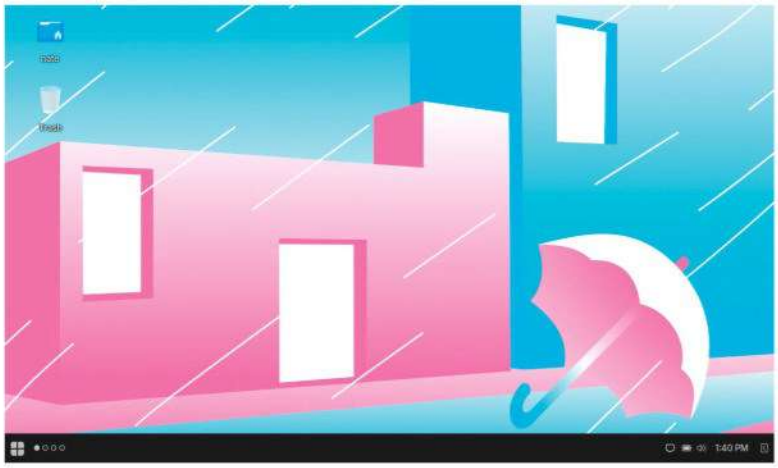
SPECS

CPU: 1GHz
Mem: 4GB (8GB recommended)
HDD: 16GB (24GB recommended)
Builds: x86_64, ARM64

Ultramarine is based on Fedora and, like its parent, follows a stable release cycle. The latest version is based on Fedora 40 and existing Fedora users can switch by way of a migration script, as opposed to installing Ultramarine from scratch.

The site's comprehensive wiki explains that the OS was created out of "frustration with the legal limits of Fedora". Being based in Thailand, the developers aren't restricted by US software patents, so can offer more packages.

The result is a more user-friendly distro than plain old Fedora. For instance, Ultramarine allows users to install packages via its own Terra repository, RPM Fusion and Flathub.



The Flagship Edition of Ultramarine uses Budgie but there are also versions that come with the Gnome, Xfce and Plasma desktop environments.

We focused on the Flagship Edition, which bundles the Budgie desktop environment, but alternative spins are available with Gnome, Xfce and Plasma. (The latter spin uses the new Plasma 6 desktop environment.)

After firing up the 2.8GB Flagship ISO, our VM encountered a kernel panic but this was resolved as soon as we adjusted to match the machine specs.

On booting into the live environment, we were impressed by the clean interface, contrasting a colourful desktop background with the default dark theme. In the Flagship Edition, you can access Budgie desktop settings to see a range of alternative themes.

Ultramarine uses the same *Anaconda* installer as Fedora. This enables you to complete setup in any order, provided you choose a suitable drive and user account. Full disk encryption is also supported.

The install took under 10 minutes, which compares favourably to Fedora and other RHEL-based distros.

Given the large ISO size, there isn't a huge number of bundled apps. Browsing is handled by the latest *Firefox (126.0).* We were surprised to see the browser contained bookmarks to the Fedora Project and Red Hat websites – but not Ultramarine's support pages.

The distro also bundles *LibreOffice Calc, Impress* and *Writer,* as well as the *Parole* and *Rhythmbox* players. You can use the integrated Gnome *Software* app to install more.

As we browsed available apps, we noted where you are offered a choice to install via the various available sources. For instance, *Gnome Reversi* is available both from Flathub and via Ultramarine's own repos.

The recommended system requirements for Ultramarine are quite weighty; it clearly isn't designed to breathe life into old hardware along the lines of Puppy Linux. Still, after firing up the pre-installed system monitor, we found that at rest the Flagship Edition consumed just 2GB of RAM in our VM. Post-install, the virtual hard disk took up only around 6GB, barely a quarter of the recommended size.

The desktop apps also opened zippily, which we used to good effect to download a 1GB ZIP file via the *Firefox* browser. System resources barely moved as we did so, with RAM usage only spiking by a few megabytes as we opened the *Zsh* interactive shell to extract the archive.

Ultramarine did struggle slightly when we fired up a deathmatch in *Xonotic*, with RAM usage creeping to 81% (around 3.3GB). Given that the recommended minimum is 8GB, we aren't knocking off points for this.

The overall philosophy of Ultramarine is stated to be an OS that "just works for new and advanced users". In this much the developers have succeeded, given how easy Ultramarine is to set up and install. The inclusion of extra packages, including proprietary drivers and Flatpaks, also gives users more flexibility over plain Fedora. LXF

VERDICT

DEVELOPER: Fyra Labs
WEB: https://ultramarine-linux.org
LICENCE: Mainly GPL

FEATURES	7/10	EASE OF USE	8/10
PERFORMANCE	7/10	DOCUMENTATION	8/10

While we'd have liked a welcome screen pointing to the excellent Ultramarine wiki, this OS is a breeze to set up.

» Rating 7/10

Linux Lite 7.0

Nate Drake goes down under to discover this distro for recovering Windows users. There's nothing wrong with a little hand-holding...

IN BRIEF

Like New Zealand, Linux Lite is beautiful, friendly and definitely worth a visit. Every effort has been made to help Windows users gently transition, although sometimes the OS oversimplifies things a little.

SPECS

CPU: 1GHz (dual-core)
Mem: 768MB
HDD: 8GB
Builds: x86_64

Linux Lite enjoys the distinction of being the only current desktop version of the OS developed in New Zealand, as per Distrowatch.

The project goal is "to make the transition from Windows to a Linux-based operating system as smooth as possible". By way of a sweetener, the main download page lists Linux Lite's modest system requirements relative to Windows 11.

Upon downloading the 2.8GB ISO, we noted that this distro uses Xfce. Clearly some thought has gone into making Linux Lite easy to master for Windows users, given that the task bar contains an easy-to-navigate menu.

Linux Lite attempts to smooth the transition from Windows to Linux as much as possible for users by listing identical or equivalent apps.

The bundled apps will also please Microsoft migrants. The inclusion of the *Chrome* browser is a logical choice, given its popularity with Windows users. Email is managed via *Thunderbird,* while media playback is effectively covered by *VLC Player.*

While *LibreOffice* is pre-installed, the Office menu doesn't use traditional names – for instance, *Writer* is listed as Word Processor and *Calc* as Spreadsheet.

This kind of hand-holding also translates into Lite Welcome, which has been slightly tweaked for the latest release (code name Galena). From here, you can install updates and drivers, as well as install extra languages. You can also switch between the default light Materia theme to something darker.

Lite Welcome also offers access to the online help manual, as well as the user forums. Our only gripe here is that new posts were listed first, meaning we had to wade through two pages of comments to view the latest release announcement.

This is where we discovered the Welcome slides had been revised, so we decided to fire up the installer. Again, the emphasis was on how simple the switch from Windows to Linux can be. The slides listed popular Microsoft programs for which there are Linux equivalents, such as *Skype* and *Dropbox*, as well as pointing out that *Wine* can run certain Windows apps.

True to its name, the install process was extremely rapid (less than eight minutes in our VM). We next decided to launch *Lite Software,* which prompted us to update the repos. Linux Lite is based on the latest Ubuntu LTS release (in this case 24.04). Still, the number of packages offered is a carefully curated list, including Windows favourites, like *Zoom* and *Skype,* as well as media software such as *Kodi* and *Audacity.*

We selected *Skype*, which installed in moments. Although it wasn't a listed app, we were also able to install the *Dropbox* daemon via the terminal, though this begs the question whether a former Windows user would know this workaround. The OS muddied the waters further by listing *Dropbox* as File Synchronizer in the program menu.

We also used the terminal to install *Gnome System Monitor.* Even at rest, the desktop was consuming a little over 1GB of RAM. This is impressive for a modern OS but is at odds with the stated system requirements of 768MB. RAM usage shot up to 2.6GB when playing Nate's favourite FPS *Xonotic* but gameplay was smooth, even when running alongside other apps.

Post-install, our virtual hard disk took up just over 11GB, which again is slightly more than as listed in the official documentation (8GB). By comparison, our clean install of regular Ubuntu 24.04 is 7.1GB.

Still, these stats don't translate into a sluggish system. *Chrome* launched in a split-second and began downloading a ZIP archive. We were then able to extract this seamlessly in the *Thunar* file manager, which now comes with a split view. LXF

VERDICT

DEVELOPER: Jerry Bezencon
WEB: www.linuxliteos.com
LICENCE: Mainly GPL

FEATURES	**8/10**	**EASE OF USE**	**9/10**
PERFORMANCE	**8/10**	**DOCUMENTATION**	**8/10**

If you have a friend who uses Windows and wonders what all the fuss is about, **Linux Lite** is one to recommend.

» Rating 8/10

Gnoppix 24.4.15

Nate Drake drills down into the newly resurrected Gnoppix – a Debian-based distro boasting great privacy features and AI integration.

IN BRIEF

Gnoppix tries to be all things to all men, offering advanced AI tools as well as privacy features. Both the setup and interface are slick but AI functionality is limited.

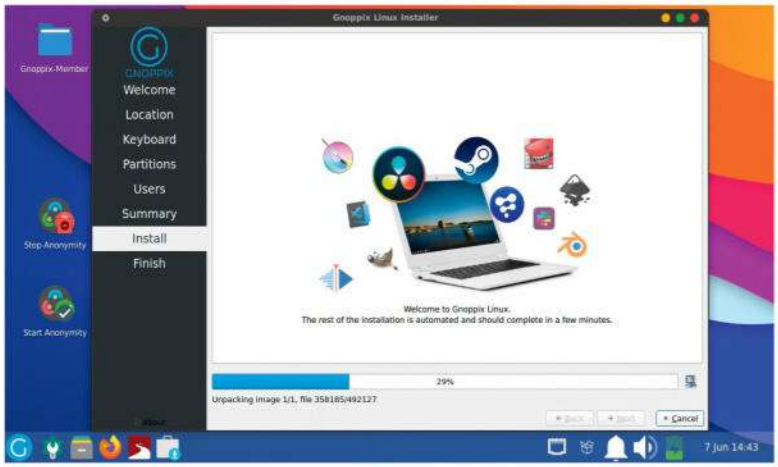
SPECS

CPU: 3GHz
Mem: 2GB
HDD: 20GB
Builds: i386, x86_64

Gnoppix was first released in 2005, though it's only in the past few years that stable versions have been regularly made available for download.

Prior to version 2022, the distro was based on Knoppix – a Debian-based distro optimised for running in a live environment. Like Knoppix, Gnoppix is also designed to load into RAM. In fact, the main developer Andreas Mueller was hired by Canonical to import this live functionality into the very first release of Ubuntu in 2004.

He discontinued work on Gnoppix, only to resume again during the pandemic. The current version is based on the latest stable release of Debian (Bookworm).



Setup is a breeze with Calamares. The pictured desktop shortcuts can route connections through the clear or dark web respectively.

According to the project website, the OS is designed for privacy while also incorporating AI features. Given the controversy surrounding Windows' Recall feature, we were keen to peep under the Gnoppix hood.

We chose the flagship Community edition of Gnoppix, which uses the Xfce desktop. There's also a subscription-only Pro release, as well as various beta versions using alternative desktops, such as KDE.

On first boot of the hefty 4.6GB ISO, you are offered multi-language options as well as an experimental persistence mode. As the desktop loaded, we were treated to the comprehensive welcome guide. From here you can find Gnoppix online support and forums.

The latest release announcement stated that this version of Gnoppix (code name Domination) bundles *Tor* to allow automatic connections to the darknet. This can be accessed via the two desktop shortcuts to Stop or Start anonymity. (By default, this is disabled.)

Though there are references to an AI installer on Distrowatch, on first launch it seems to be just the regular *Calamares* setup assistant. We used this to install Gnoppix in a virtual machine, though *Calamares* vanished while unpacking images. After restarting, Gnoppix successfully installed after seven minutes.

Upon rebooting, we were presented with an error saying the default Qogir GTK theme had failed to load. The Load Fallback Theme button failed to produce results. One reboot later and the desktop loaded.

We fired up the elegant Whiskermenu and clicked into Gnoppix AI. This launched a web page that explained that the OS has its own uncensored LLM Gnoppix GPT – but only for members. As Nate's application for membership was still pending approval, we launched the Gnoppix AI installer instead. This opened a virtual store of verified AI *Pinokio* scripts. We attempted to use this to segue around being excluded from the Gnoppix club by downloading *SillyTavern*, which allows users to interact with other LLMs, but the install failed due to a network connection error.

When we launched *Gnoppix Diffusion,* a script was executed to download a huge number of WHL archives and Python scripts. Many gigabytes later, *Easy Diffusion* launched in the default browser, but after an hour was still processing a sample image.

We had more joy with *Gnoppix Software Manager,* which in theory can be used to install more mundane apps. We attempted to install the Flatpak version of *LibreWolf*, only to be told that our administrator password was invalid. After using *Terminal* to set a new password for root, *LibreWolf* installed successfully, albeit rather slowly.

Naturally, there was no need to do this. Gnoppix bundles *Firefox 115.10 ESR,* which incorporates *uBlock Origin* and *Perplexity AI.* Other pre-installed apps reflect Gnoppix's emphasis on security, such as privacy-friendly email service Tuta and *Proton VPN.* LXF

VERDICT

DEVELOPER: Andreas Mueller
WEB: www.gnoppix.com
LICENCE: AGPL 3.0

FEATURES	7/10	EASE OF USE	6/10
PERFORMANCE	7/10	DOCUMENTATION	7/10

Gnoppix has an excellent and rich UI. It's also easy to set up, with some privacy-friendly tools. Shame about the AI.

» Rating 7/10

Berry Linux 1.39

Nate Drake goes after a taste of the Fedora-based distro Berry Linux but finds it's more sour than sweet.

IN BRIEF

While its minimal apps and lightweight size mean it lives up to its promise of being lightning fast, there are too many bugs and glitches to justify using Berry Linux.

SPECS

CPU: 1GHz
Mem: 128MB
HDD: 2GB (optional)
Builds: i686, x86_64

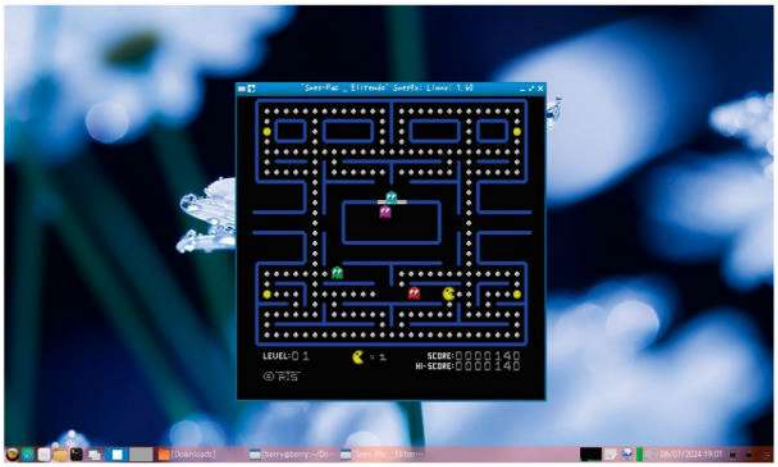
Like Gnoppix, Berry Linux was once based on Knoppix. These days, it's based on the latest version of Fedora (in this case, Fedora 40).

This distro's USP is that while it does support persistent installation to USB, it's primarily designed to run as a live distro. Both ISO and IMG files are available, though when we visited the site's download page, there were no direct links to the latest version (1.39).

After some scrolling, we happened upon a SourceForge link (**https://sourceforge.net/projects/berryos/**) that contained the latest release in the Files section.

As it was created by a Japanese developer, it's available in both Japanese and English. On first boot, you are offered a choice of languages.

The project website describes Berry as both lightweight and lightning fast. This is borne out by the minimal CD-sized ISO (714MB), plus the fact that the desktop loaded in less than three seconds.



A spot of "**waka waka**" is **a welcome** distraction from **the** infantile default fonts **and** inability to install **extra** programs.

No welcome mat

After reviewing so many user-friendly distros this month, we were disappointed not to see a helpful welcome screen for first-timers to get started with the OS. This was unsurprising, though, as the main website also has no obvious links to a wiki or user forums.

Still, the developer has clearly put a lot of work into this distro, given that it has its own display manager *(berry-dm).* Nakada-San has also included *Miss Driller 2,* a Berry-specific clone of *Mr Driller.* We found this in the Games category, alongside a SNES emulator with a playable version of *Pac-Man.*

Berry's default browser is *Firefox* but it also implements *Ice SSB,* to allow users to launch web apps from the main menu. These include the online versions of *Microsoft Word, Excel* and *PowerPoint.*

Besides the *Slypheed* email client and a CD-ripping program, this is pretty much the extent of Berry's third-party apps. As the OS has no graphical package manager, we fired up the terminal (both *Sakura* and *Xterm* are present). Unfortunately, when we tried to run `dnf install` to update and add new packages, the utility failed to connect to the necessary repos.

We were able to use the terminal, however, to run `free` to check RAM usage. At rest the OS used just under 2GB. While the distro remained lightning fast, this is far more than the 128MB in the documentation.

Like Fedora, Berry Linux uses the *PCMan* file manager, but we found this buggy. On downloading a ZIP archive, clicking into Extract To or Extract All failed to produce any response in the context menu. Admittedly, the archive opened readily in *File Roller.*

We had far more success configuring the desktop from system settings. If you're unhappy with the drab default wallpaper, there are 17 alternatives, including pictures of adorable fluffy kittens. From here you can also change the font for label text. You need to launch the *Openbox* configuration manager if you want to change window titles from the childish default font.

Naturally, in live mode these changes aren't saved beyond the next reboot. We attempted to launch the tool to 'Save Berry Configuration' from the app menu. This resulted in an error saying it had failed to create a desktop shortcut. The same error occurred when we tried to access network card configuration.

Normally in these circumstances we'd take to the user forums or online documentation to see if this error has occurred before, but there are none. In fairness we imagine Nakada-San has his hands full with single-handedly developing and maintaining the OS. LXF

VERDICT

DEVELOPER: Yuichiro Nakada
WEB: https://berry-lab.net
LICENCE: GPL

FEATURES	4/10	EASE OF USE	7/10
PERFORMANCE	5/10	DOCUMENTATION	1/10

The Japanese proverb 'A **berry falls** to its roots' **doesn't apply.** **Unlike** Fedora, **Berry** is **buggy with minimal documentation.**

» Rating 4/10

Capes

Management is delirious at the thought of super-powered workers who can do more and be paid less. **Robin Valentine** might use his powers for evil…

SPECS

Minimum
OS: Ubuntu 18.04 64-bit
CPU: Quad-core
Mem: 8GB
HDD: 4GB
GPU: Nvidia 960 GTX, AMD RX 590

Recommended
OS: Ubuntu 20.04 64-bit
CPU: Quad-core
Mem: 8GB
GPU: Nvidia 1070 GTX, AMD RX 5700

All the best superhero teams have one thing in common: everyone has their role. While Wolverine's charging in, claws flashing, Cyclops is up on a ridge shooting down lasers, Iceman is zipping around freezing foes, Jean Grey is delving into a supervillain's mind, and Colossus is body-blocking energy blasts.

That's what your squad of superheroes feels like in *Capes* as you direct them in turn-based battles. From teleporter Rebound to psychic Mindfire to speedster Mercurial, each character is wonderfully distinct, and securing victory means combining their unique powers to enhance their strengths and cover their weaknesses.

Superheroes, including **speedster** Mercurial, **battle** thugs.

The 'Capes' are not typical superheroes, however. They're based in King City, a near-future metropolis that's been ruled by supervillains for decades. Under their oppressive regime, powers are outlawed, with any civilian showing signs of having them rounded up for a grisly fate. Your crew is less like a classic crime-fighting team and more like a resistance cell – young supers who have managed to escape capture, fighting back against the dystopian government to try to free their home.

The path to revolution consists of a series of story-driven turn-based strategy missions – always a punch-up of some kind, but often with a twist, such as stealth elements, a race against time, or civilians in need of protection. They're not easy; it's the kind of game where when you first load up a mission and see an absolute horde of enemies arrayed against you, it's natural to think, "How is this even possible?" Half an hour later, after several attempts, you finally stand surrounded by defeated thugs thinking, "How did I pull that off?"

The key to victory is always in getting the most out of every member of your team. Their identities are extreme by design. Facet, for example, clad in crystal armour, can take assault rifle bursts without flinching, while Rebound can be downed with just a few punches – but Rebound's ability to teleport gives her mobility unlike anyone else, and her backstab attacks do enormous damage. When together, Facet's abilities become a toolset for keeping her out of trouble – taunting foes, trapping others with crystal growths, and protecting her with crystal shields keeps her safe enough to get in position to strike targets.

With eight heroes available to unlock (you usually choose four to bring into each mission), and increasingly dangerous enemies arrayed against you, the web of possible combos becomes a wonderfully layered combat puzzle. All sorts of complementary mechanics add to your options. Team Ups, for example, let heroes directly combine their powers if they're physically close enough, unlocking new versions of abilities. Use Mercurial's Dash while she's near the pyrokinetic Ignis and she leaves a trail of flame behind her; backstab a target with Rebound when Mindfire's close and he mind-controls them into turning around to ensure she deals maximum damage.

Each hero also has a different way of charging Ultimate energy, the juice they need to unleash their most powerful attack, further encouraging you to commit each character to their role. Facet, for example, is such a dedicated tank that he charges up

A battle against security forces and super-soldiers.

every time he gets hit, whereas the storm-calling Weathervane gets a charge for every extra target he's able to catch with his chain lightning attack. Finally unleashing an Ultimate – setting Rebound off on a teleport frenzy where she backstabs everyone in sight, or unleashing a huge storm with Weathervane – is a lovely mid-mission dopamine hit, rewarding you for getting the best out of that character.

No strategy game would be complete without a ton of upgradable stats and capes.

Super troopers

Between missions, heroes can enhance their abilities back at base, and you're free to replay past missions, hoping to complete more of the bonus objectives, allowing you to grind for level-ups if you wish. But crucially, upgrades never flatten the differences between the characters. Rebound, for example, has three health – no matter how good she gets at teleporting and takedowns, she never gets any tougher, even as enemies become increasingly numerous and dangerous. That means that she's only a couple of blows away from death from the start of the game right to the end, keeping her hit-and-run play style intact throughout.

It's these kinds of clever decisions that keep *Capes'* battles brilliantly tactical and riveting all the way through. Great pacing helps there, too – when you're feeling a little frazzled after a particularly complicated boss fight or a careful stealth mission, that's when the game throws you a big dumb fight against an overwhelming horde and lets you unleash satisfying havoc on them. It almost never lets up on the challenge, but developer Spitfire Interactive is great at tickling different parts of your strategic brain as you play, rather than hammering on the same one to the point of frustration.

Where the pacing unfortunately isn't quite as deft is in the story. There are a lot of cutscenes in *Capes,* but very few of them really go anywhere – act two of three takes up the majority of the run time, and consists of far too much meandering around without any clear idea of what the characters are trying to achieve, before suddenly act three brings things to an abrupt close.

Broad themes of capitalism gone awry and systemic injustice work well enough as a framework, but individual story threads within that often feel incongruous. There's much agonising from the characters about whether it's OK to kill supervillains or not, for example – that's an overused superhero trope at the best of times, but in a story about a violent insurrection told in the medium of lengthy turn-based fights, it doesn't fit at all. You take down 30 thugs by electrocution, setting them on fire, and hurling chunks of concrete at their heads, only for Rebound, whose fighting style is entirely based around sticking foot-long knives into people's backs, to give you a lecture about when lethal force is appropriate. It's absurd, and only becomes more so as you discover quite how irredeemably evil the main supervillains they're so concerned with sparing actually are.

Unfortunately, lacklustre presentation doesn't do the story any favours either. There's a plastic quality to the characters, which combined with stiff animation, makes them look like action figures. In fights, that's fine – the game visually conveys what it needs to make combat run smoothly – but frequent cutscenes zoom in for face-to-face chats, forcing you to see every visual flaw as the heroes emote unconvincingly to inconsistent dialogue.

Really, though, it's not the story that pulls you along – it's the depth of strategy on offer. The drama isn't in the cutscenes, it's in every nail-biting turn as you try to figure out the perfect combo to get you out of another fine mess. It can be a serious challenge – if you're the kind of person who hates the idea of reloading the same mission multiple times looking for the right approach, this probably isn't for you. But meet it on its own terms and *Capes* will give you 20-30 hours of wonderfully substantial and engrossing superhero strategy. LXF

Facet and Rebound talking during a cutscene in Capes.

VERDICT

DEVELOPER: Spitfire Interactive
WEB: www.spitfireinteractive.com.au
PRICE: £33.99

GAMEPLAY	8/10	LONGEVITY	7/10
GRAPHICS	6/10	VALUE	7/10

An awkward story doesn't hold Capes back from being one of the best turn-based strategy games of recent years.

» Rating 8/10

Roundup

Jami » Element » Telegram » Signal » Discord

Michael Reed is a chatty sort of bloke, mature but sometimes secretive, and always secure, like a good chat app.

Secure chat services

When chatting, we want to know that our secrets are kept safe, so **Michael Reed** is taking a look at five security-conscious chat systems.

HOW WE TESTED...

We installed each chat client on to our Linux Mint test machine and an Android phone. Where possible, we also tried out the web client using *Google Chrome*. Once installed and signed in, we added some contacts and a number of groups.

We feel that this is a good way of using these systems – taking advantage of a computer setup, when available, and swapping to mobile on the move. We looked for any synchronisation issues as we swapped between mobile and desktop platforms, but we didn't run into any problems with any of the systems.

We prefer to examine areas of a system where there is significant distinction. Video chat and voice chat are important features, but our finding was that all five systems had fairly comparable facilities in this area, including group video conferencing.

If people like cosmetic features such as stickers and purchasable emojis, good for them, but we've decided not to delve too far into this area.

CREDIT: Getty Images

We're looking at applications that enable you to securely send messages to individuals or groups using your phone or your Linux-powered computer. If you're an MI6 operative, a software developer who needs to collaborate or simply someone who needs to organise a trip to the pub, you could well make use of one of these apps.

As well as being useful one-to-one tools, some of these apps can give access to social network communities that you might have been unaware of. Even better, all the systems we're looking at offer voice and video chat.

Whether it's the government feds or the megacorps that want to know your every move, we're big fans of chat systems that keep things secure and private thanks to the latest encryption tech.

We're a bit biased in favour of the needs of the typical Linux user here, as we like applications that enable you to move seamlessly between taking part in a discussion on your computer to chatting on a mobile device. Desktop usage gives the benefits of a big screen and keyboard, while mobile usage enables you to take the chat with you wherever you go.

Manage groups

Converse with coworkers or hook up with like-minded folk to chat.

As well as individual-to-individual messaging facilities, each of these chat systems contains at least some support for groups. Group facilities can mean community chat based around common interests or business discussions for a given workgroup.

Telegram group chats benefit from the popularity of the overall platform, and we were able to find plenty of groups concerning topics such as Linux, current events or hobbies. Its popularity in the world of group discussions means that Telegram is a social network in its own right. The features for private chat among groups that you create or join are strong, too.

Gamer-driven group chat is the first thing most people think of when they think of Discord, but the groups that exist are plentiful and cover every possible subject, such as medical discussions, Linux software, Linux distributions and software development. The group chat features are extremely good, and a server, as it is called, can have subchannels to separate the chat into more specific areas (such as an 'introductions' section). Overall, it's the best group chat feature we've seen. There's nothing stopping you from using it in a business setting, but Discord might not be embraced in a corporate environment due to its reputation as a gamer-focused tool.

Signal has group facilities that are usable for business teams or family groups, but we couldn't find a lot of public discussion groups on general topics. The same is true for Jami, and this is a downside of its lack of mainstream popularity as a chat system.

Discord's group chat features are excellent. Channels exist within a main group so that it isn't overwhelmed. This is an example server for the Sunshine streaming tool.

Element sits atop the Matrix system, which means it has access to chat groups orientated around topics such as open source software, and quite a few open source developers use this network. The group chat system supports a topic threading system, but it's hit and miss in terms of identifying the threads.

VERDICT

JAMI	5/10	SIGNAL	5/10
ELEMENT	7/10	DISCORD	10/10
TELEGRAM	7/10		

Discord is the clear leader when it comes to the popularity of its group chats. It can be used for business, too.

Linux installation

Setting up the all-important Linux client to get things started.

We could have installed Telegram straight from the Linux Mint repository, but that version was fairly out of date. The website hosted links to the Flatpak and Snap packages, and we went with the Flatpak.

The Element website had instructions for Linux installation, which amounted to adding a custom repository for Debian-derived distributions. This is a bit limiting, and for this reason, we used Flatpak to install Element, even though the Snap package was also fully up to date.

Much the same could be said for the Signal installation situation. The official website gives instructions to add a custom repository that is suitable for Debian-derived distributions by cutting and pasting commands. The Snap and Flathub repositories had fully up-to-date versions, so we installed from Flathub as it was easier. Like Telegram, Signal cannot be set up without a smartphone with a valid number.

Jami also offered a combination of custom repos; however, unlike some of the others, it had support for most of the major Linux distributions (and different versions of each distro) when doing this. The site also advised on Snap installation. Although Flathub had a version of Jami, the Snap version was slightly ahead of it, so we used that. We were prompted to manually enable ALSA access, for sound support, at the command line.

As often as not, we were able to install an up-to-date version of a chat app by selecting the Flatpak version. Here, we did so using Linux Mint's Software Manager.

Discord is available on Flatpak, but has some limitations due to sandboxing, so we installed the DEB from the official site. There's also a TAR.GZ that can be adapted to distros that are dissimilar to Debian. Desktop Discord has its own update checker routine.

VERDICT

JAMI	10/10	SIGNAL	7/10
ELEMENT	7/10	DISCORD	7/10
TELEGRAM	7/10		

Flatpak or Snap ended up being the easiest installation options. Jami had the broadest distribution support.

Desktop client apps

Software to access the service.

All of the systems that we're looking at here provide a minimum of a mobile phone app (iOS and Android) and desktop applications for all of the major platforms, including Linux. We expect the mobile app to be fast and easy to use when carrying out the core functions of single-user and group chat, although no on-screen keyboard can be as good as the real thing.

All of the Linux clients work in a similar way. They install as a desktop application and then provide a windowed environment that offers a similar interface to the mobile phone application.

Naturally, we expect a few benefits from running the desktop version of the app compared to the mobile client, beyond the fact that we have ready access to a real keyboard. More elements should be available on screen at the same time. In the case of either mobile or desktop versions of the app, if it's a nice-looking environment, that's a bonus.

Jami 7/10

On Linux, it's not yet the default, but we enabled the seamless mode, as it seems like the way forward for Jami. Once you've done this, everything has a clean, light, modern look to it. Beyond this, there are System, Light or Dark themes, along with a zoom factor, meaning that all basic usability situations should be covered.

The message bubble format takes up extra space and favours text-style messaging rather than deep discussions. The whole thing takes place within a two-pane display, with contacts in a column on the left-hand side of the window.

We would make the same comments about the mobile app. The home screen displays the contacts, and a single press takes you to the actual chat screen. On desktop or mobile, it's a simple, easy-to-use interface that looks good.

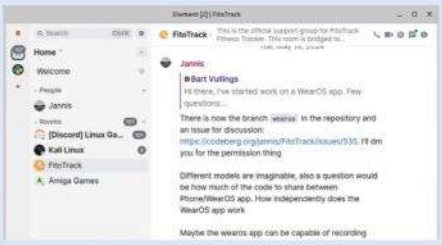
A web client has been under development for quite a while, but it hasn't been released yet.

Element 9/10

The Linux application defaults to a system theme, but there is a light and dark mode, too. We liked the way that you can specify message bubbles or an IRC-like style that lends itself better to detailed discussions. This reflects the nature of the community group chats that are available through the Matrix network that Element uses. There are essential basic theme settings, control over interface zoom level and detailed user interface settings, and it supports Markdown in chats.

The Android app also benefits from both styles of on-screen message presentation, and you'd probably settle on one or the other depending on the type of chat or messaging you prefer.

It's always great to have a web option, and the Element web client experience is near-identical to the desktop app, so the choice between the two depends on whether you prefer to have it running in a window or a browser tab.



Licence and freemium features

Is the free package usable, or is it worth paying extra?

The basic version of Telegram is free to use, but there is a paid tier called Telegram Premium that generally doubles things, such as the upper limit for subscribed groups, but most of the limits are reasonable on a free account anyway. The Premium version also removes the advertising in public channels, offers voice-to-text and an improved interface for chat organisation. In our opinion, the free account should be sufficient for the majority of users, including business users.

Signal is completely free to use and there are no paid tiers at all. There are no adverts, and it's possible to donate to the project. All of the clients are open source, and the server is open source as well. This means there are no limits to file sizes, bandwidth or number of users.

Discord isn't open source at all, but it's free to use. There is a paid tier called Nitro that extends the limits of how many servers (groups) you can be connected to, with additional visual bling thrown in. In addition, it's possible to set up a group, limit access to it and monetise it. We consider the basic, free service to be adequate for most users.

Element is free to use, and it's open source software in terms of its clients and server. You can use the publicly available Matrix.org server or self-host, and there are various paid server options.

Jami is much the same in that the clients and the server are completely open source. The official website offers an enterprise package that allows logins and management using corporate credentials.

VERDICT

JAMI	9/10	SIGNAL	9/10
ELEMENT	9/10	DISCORD	7/10
TELEGRAM	7/10		

We haven't considered any options that don't offer a useful free service, but we prefer the open source options.

Telegram 7/10

The mobile phone app is slick and highly responsive. It's not entirely plain looking as some of the other messenger apps are, but there's no real UI cruft that gets in your way. Simply click on a contact or a group, and you're ready to start chatting.

Particularly when engaging in publicly available chat groups, the content tends to be different from that of some of the other services, imposing a different visual style. The message bubbles with a set of reaction icons suit social-media-style streams of posts or news items with a graphic attached. It wouldn't be our first choice for detailed serious discussions, though.

On the mobile app, everything seems perfectly proportioned for some scrolling or chatting. On the desktop app, there's more room to stretch out the layout, but it doesn't quite have a glove-like fit as space feels wasted on a desktop display compared to Facebook, for example.

Signal 6/10

The Signal desktop application has what amounts to a two-pane interface, and it's a neat, functional affair that sits within a normal app window. Appearance options cover the important basics, such as light/dark colour schemes and an overall zoom factor, and the zoom can be adjusted from a menu option or keyboard shortcut.

For the chat itself, message bubbles are used rather than single lines of text, which suit the chat style of interaction more than discussion, and it's certainly more at home on mobile than desktop.

If anything, the mobile app looks even more inviting as it employs a little more colour in pastel shades of some of the floating buttons. In the case of the desktop or the mobile app, it's quite likely that many Linux users will appreciate the understated, but stylish, look.

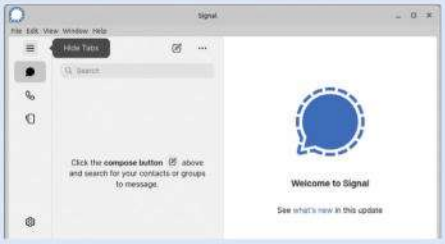
Signal doesn't feature a web client at all, apparently because it could compromise the security of the platform.

Discord 8/10

Discord is right at home on the desktop, and the experience when using the Linux client is roughly the same as when chatting via the web version. They both make good use of a full-sized screen. In terms of overall layout, you can always see the available groups, the channels in that group and the user list, along with the main text area.

By default, the theme is dark grey, but the light theme is also highly usable. Rather than placing each line of text into a bubble, the text is presented plainly, which is more appropriate for detailed discussions. We can see why Discord is a favourite of open source developers.

Things can't be as spread out in the mobile app, of course, but the workflow between choosing a contact or a group before moving on to the actual chat is good enough. Aesthetically, the mobile app is slick looking, with panels smoothly sliding in and out as needed.



Popularity stakes

Making sure there's someone on the other end of the connection.

Not quite matching WhatsApp's 2.4 billion users, Telegram is no slouch with around 500 million users. It's practically a social media network unto itself, with loads of interesting groups that you can interact with via the web, desktop and mobile apps. It's the most popular of the apps that we've looked at here.

With over 150 million users, Discord still falls within the mainstream. It tends to be used a bit differently from some of the more popular systems because it started life as a communication tool for gamers. The mobile app is very good, but the web and desktop clients are a more popular way of using the service. If the people you want to interact with don't already use Discord, it may be an easy sell to convince them to create a Discord account and use it via the web.

Signal doesn't have the instant brand recognition of something like Telegram, but it's quite a popular service, with around 40 million users, largely due to its online reputation for excellent privacy and security features. On the downside, though, there is no way of setting up an account without a mobile phone.

Element isn't that well known in mainstream circles, but the default network that it connects to, Matrix, is highly popular. Indeed, it's worth having an account at the ready because it is the best first point of contact with some open source software communities and developers.

As good as Jami is, it doesn't come within the top 20 or so messaging apps in terms of popularity, and you may have to persuade people to add it as an extra messaging app.

VERDICT

JAMI	5/10	SIGNAL	7/10
ELEMENT	7/10	DISCORD	9/10
TELEGRAM	8/10		

Telegram's popularity helps when persuading people to install it and means that there are useful communities already established on the platform.

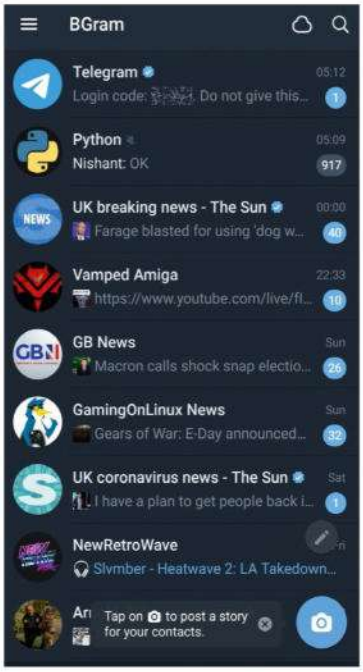
Alternative clients and extensions

Can you persuade your chosen system to work a bit differently?

Discord is a largely proprietary system, but the API is an open one and this means that there is a surprising number of open source clients and utilities, largely thanks to the popularity of the service. There are at least half a dozen complete replacement clients, and most of these claim to offer better privacy than the stock Discord client. There are quite a few extensions available for the official desktop and web clients that add small convenience features, such as hiding disabled emojis, showing user permissions in more detail and integrating Google Translate into the interface.

Telegram has open source official clients, and there are quite a lot of unofficial ones, too, and this includes unofficial mobile applications as well as those for the desktop. The Telegram extension system consists of bots that can automate responses to customer enquiries, and a number of browser extensions that add extra features when accessing via the web app.

As it's open source, the Signal client could be forked, but there isn't much support for this within the developer or user community. Part of the appeal of the system is that everyone is using the same, certified official clients to access the service.



BGram is an unofficial mobile client that uses the Telegram API and network. It's got its own UI and adds a few features such as integrated support for the secure Tor network.

Element is just one client that uses the Matrix network, and the Matrix.org website has a list of other clients (**https://matrix.org/ecosystem/clients/**).

Jami has an extension system that works for the desktop or mobile clients. Five official extensions are available on the website that do things such as add subtitles to voice chats.

VERDICT

JAMI	6/10	SIGNAL	3/10
ELEMENT	7/10	DISCORD	8/10
TELEGRAM	7/10		

Surprisingly, Discord has attracted a vibrant ecosystem of alternative clients and add-ons.

Security model

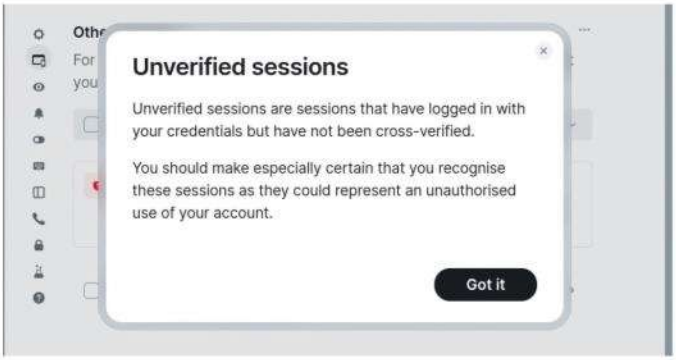
Feds, cyber crims, megacorps - let's make sure they can't get in.

Jami is peer to peer without a centralised server. It uses end-to-end encryption and is open source, so experts can examine the code. Not being based around a server has advantages, but limits functionality in terms of finding users.

Discord relies on a central server, connected to by desktop and mobile clients. The inner workings of the clients and server are secret as the source is closed. Discord's popularity means it's a target for hackers, and the server has been compromised more than once. When you delve into the public group chat servers, you start to lose control of who you're rubbing shoulders with.

Element uses end-to-end encryption and is open source, both marks in its favour. Rather than using a centralised server, it's decentralised using the Matrix protocol. For free, you can use the Matrix.org server to organise this or you can self-host your own server. Paid servers are also available. Your device isn't regarded as secure until you have verified it via another of your devices or your computer, another layer of security. This adds up to a system that's near-impossible to intercept by governments or criminals.

Signal, on the other hand, does make use of servers, a possible point of failure for security for some systems. But, because of the strong encryption that is employed, even if the head of the NSA was sitting in the server room, it's virtually impossible that your messages could be compromised. The open source nature of the server code means that it can be examined by security experts. It's a highly secure system overall.



Element insists on verifying a desktop, web client or mobile app. It offers extra security, but make sure your mobile client is verified before you leave the house.

Some of Telegram's inner workings are secret because the server is closed source, even though the clients are open source. It doesn't use full encryption at every stage unless you invoke a secret chat. It's fairly secure, but there's a small chance that hackers could break into the server or intercept your messages.

VERDICT

JAMI	9/10	SIGNAL	9/10
ELEMENT	9/10	DISCORD	5/10
TELEGRAM	6/10		

Jami, Element and Signal all offer a secure operating model, which is part of their overall appeal.

The verdict

Chat applications

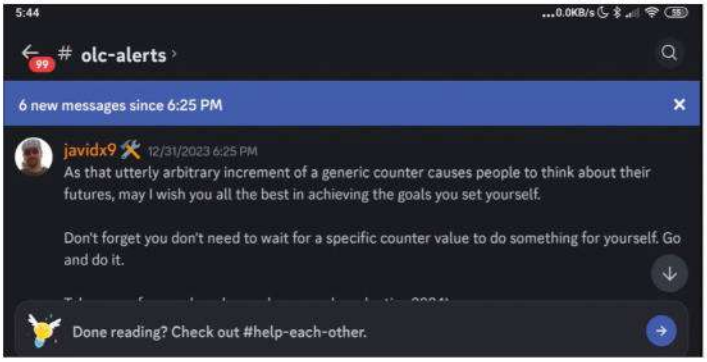
Discord has a bit of a reputation problem as it's so strongly associated with gaming. However, it can be used as an instant messenger and a group chat application in pretty much any setting, including serious ones such as business collaboration or software development. Well-organised community chat in text mode is the heart and soul of Discord. For this reason, Discord is one of the most popular hangout tools for Linux software projects. It might be because we're nerds, but we prefer the IRC-like feel of the text chat for when we want to get into serious discussions. The facilities are also there when you need to go one-on-one with text, voice or video chat.

Telegram had the opposite evolution to that of Discord because it's a mobile phone app with perfectly usable web and desktop clients, rather than the other way around. Its popularity means that many people who you would want to communicate with may already have it. Or, if not, they should be easily persuaded to install it. What makes Telegram more than just a secure messaging app is the sheer number of groups that use it. For example, it's handy to have it installed on mobile with some interesting news site feeds to scroll through when time permits.

Element may not be quite as well known as some of the other secure messaging apps, but the Matrix network means that it's quite often a good choice for setting up an open community discussion group or connecting to an existing one. For most users, the facilities in the areas of instant messaging and voice or video chat are solid, too.

Signal could easily have fallen into the trap of being an excellent open source solution that only the experts know about, but it's so good at what it does that it has quite a following among those who want the basic facilities for group and individual chat. Its security model means that all of this can be carried out without prying eyes getting a look in.

Jami is in a similar position to Signal and stacks up similarly in terms of features. However, it doesn't have the same user base, which might be an issue in some circumstances. The fact you can sign up without a phone number is a particular mark in its favour.



1st Discord 9/10

Web: **https://discord.com** **Licence:** **Proprietary**
Version: **Stable 300109 (Linux desktop)/231.13 (mobile)**
Usable for most types of chat. Strong in the area of community chatting.

2nd Telegram 9/10

Web: **https://telegram.org** **Licence:** **GPL 3 (clients)/Proprietary (server)** **Version:** **10.13.4 (Android)/5.1.4 (Linux desktop)**
Mainstream recognition. Slick interface. Good communities and news feeds.

3rd Element 8/10

Web: **https://element.io** **Licence:** **Apache-2.0**
Version: **1.6.14 (Android)/1.11.68 (Linux desktop)**
Connects to a large community. Highly secure. Good for discussion or chat.

4th Signal 8/10

Web: **https://signal.org** **Licence:** **MGNU Affero General Public Licence** **Version:** **7.8.1 (Android)/7.11.1 (Linux desktop)**
Basic personal and group chat. Clean interface. Extremely secure.

5th Jami 7/10

Web: **https://jami.net** **Licence:** **MIT Licence**
Version: **20240408.1 (Linux Desktop) / 20240529-01 (Android)**
Highly secure. Covers the basic features. Not that well known.

» ALSO CONSIDER

WhatsApp is the most obvious omission from our list of secure messaging services. We decided not to include it because it lacks an official Linux client.

There are many possible Matrix network chat apps to choose from other than Element. Fractal is such a chat application specifically coded for the Gnome desktop written in the Rust programming language, for example, and there are clients tailored to most environments and toolkits. Generally, it's worth browsing the list on the Matrix.org website (**https://matrix.org/ecosystem/clients/**).

Viber is a popular instant messaging app (it's the most popular one in Ukraine) and it offers a Linux desktop version, but it's closed source, and didn't make our list this time.

Session is open source and offers a Linux desktop app. Not requiring a phone number helps maintain user anonymity as does the decentralised blockchain network it runs upon.

Boost your VPN privacy

Jonni Bidwell emerges from his encrypted tunnel with the facts and fiction about virtual private networks...

It's almost impossible to avoid propaganda and hype about virtual private networks (VPNs). Anywhere you look on the internet there are adverts, advertorials and arguing about why you need (or don't need) one, which one is best, or how you can prove one isn't run from someone's basement. If you buy into all this hyperbole, you'll be sold on the idea that a VPN will protect you from government spies, nosey ISPs and advertising agencies. Indeed, you'll be rushing to hand over your card details because this amazingly priced deal won't last for ever.

But stay your hand. There will always be cheap deals, and they will try to bill you for ever. Sadly, all subscription services work this way. Except Linux magazines. Which you'd never want to unsubscribe, from, would you? We digress. We're here to tell you how VPNs work their magic, what you should (and shouldn't) use them for and even how to set up your own.

We'll help you navigate all of the advertising bluster. And we'll show you how to get started with a good VPN service, from the good folk at Proton. They started out at CERN, so they know their subatomic particles, and being the boffins behind Proton Mail, they know a thing or two about privacy, too. Oh, and they have excellent Linux support. If you want to shop around, we'll help you with that, too. And, of course, no **LXF** feature is complete without a healthy dose of rolling your own. So, we'll also guide you through running OpenVPN or WireGuard on your own infrastructure.

CREDIT: Magictorch

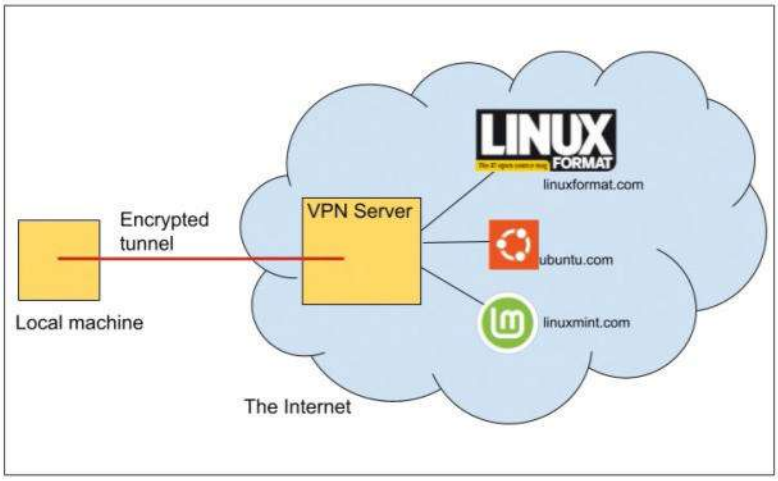
The wherefore and the why

What is a VPN and how can they afford to sponsor so many streamers?

You've probably seen adverts for at least three VPN providers on your favourite YouTube channel. No doubt the presenter will have extolled the privacy and security virtues of their benevolent sponsor (perhaps while conflating those terms), imploring you to sign up with a bespoke discount code. Maybe you signed up a year ago, have just been stung for another year at full price, and are wondering what you're really paying for (or if you should really trust that YouTube personality). Perhaps you sometimes have to use a VPN for work, and wonder how this relates to the consumer VPNs for which your favourite tech review site is peppered with referral links *[stop mocking TechRadar; the execs will cancel us – ed]*.

A VPN is a proxy that routes all or some of your traffic through an encrypted tunnel, with the effect that your ISP (internet service provider) can only see that you are connecting to that VPN. It can see how much traffic is flowing, but not where that traffic's end points are or what that traffic is (any and all protocols can be stuffed into the encrypted tunnel). Conversely, any site or resource that you access via a VPN can only see the VPN's IP address, not the one provided to your home router by your ISP. It's worth noting that there are proxy services that are technically distinct from VPNs. The main difference is a lack of encryption in the tunnel, which technically doesn't matter if you're just tunnelling web traffic from HTTPS sites. However, it's possible that a web proxy might be set up in such a way as to forward your real IP address (for example, for WebRTC traffic that needs a direct connection), which nullifies any privacy suppositions. That said, techniques such as TunnelVision *(see box)* or the long-standing iOS decloaking vulnerability (**www.michaelhorowitz.com/VPNs.on.iOS.are.scam.php**) might also have this effect.

VPNs have been around for about as long as the web itself, but were initially only used to restrict access to corporate networks. More specifically, companies wanted to allow remote or off-site workers access to their resources, but without the risks associated with putting them on an internet-facing server. Even in the early days, people knew it was a bad idea to do this, even if that machine is protected by a password or key. Instead, a public VPN server was set up and legitimate users could authenticate with this using their usual work credentials (and possibly a certificate if access was restricted to, say, company-provided laptops). Connection would be over a protocol such as PPTP (Point-to-Point Tunnelling Protocol), IPsec (standard IP packets wrapped in encryption), L2TP or, later, OpenVPN. Then, thanks to some nifty routing tricks, the remote user could access work resources as though they were in their office cubicle. Depending on the setup, the remote machine might see all of their traffic routed through the VPN (so that it was subject to the same restrictions as in the office) or non-corporate traffic being allowed to flow unimpeded.



A grossly oversimplified view of what a VPN does. The websites pictured would be oblivious to your IP address.

» TUNNELVISION

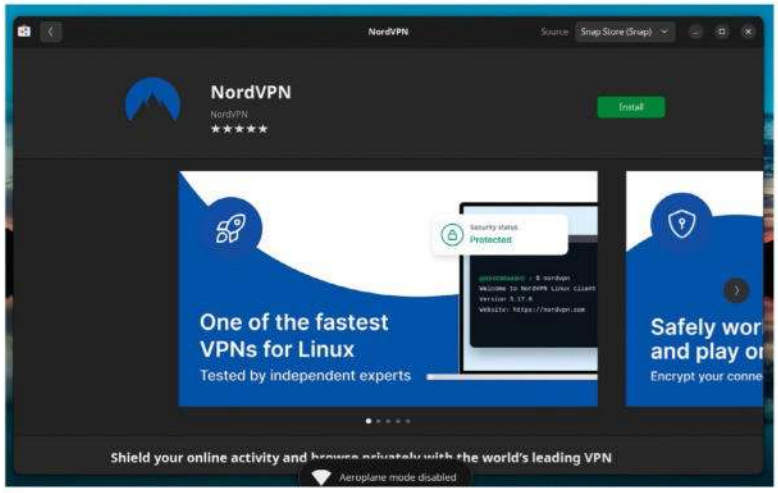
At the end of May 2024, researchers from Leviathan Security Group announced that they had discovered a vulnerability (CVE-2024-3661) that they called TunnelVision (see **www.leviathansecurity.com/blog/tunnelvision**). When exploited, it reroutes traffic outside the VPN tunnel, undoing any privacy benefits and allowing all traffic to be sniffed. What was shocking about it was that it affected Windows, iOS, Mac OS and Linux, and could potentially have been exploited since 2002. It also relied on a relatively simple trick, running a malicious DHCP server on the target's network. This was enough to thwart all modern VPN protocols, namely OpenVPN, WireGuard and IPsec. Only Android was immune, thanks to its DHCP implementation not supporting a certain parameter.

TunnelVision can be mitigated against if your DHCP server allows you to disable this setting (called 121), but the exploit can then be used to disable network access entirely. On Linux, namespaces can be leveraged to defeat the attack, but these aren't something your average user wants to be involved with. Naturally, an attacker already needs some sort of foothold on your network to run the malicious DHCP server, but this is a far lower bar than traditional exploits. It's also easy to do by setting up a public Wi-Fi hotspot. Many have said the real threat has been overestimated, but it is food for thought.

Good VPNs and bad

There may be more to consumer VPNs than meets the eye...

In the 2010s, consumer VPNs began popping up, and a whole new anxiety-driven market emerged, based largely on half-truths and people's misunderstanding of how networks work. VPN providers promised their users heightened privacy, a way to circumvent geoblocking (sites that only allowed certain content to be viewed from a particular region) and all manner of other nebulous benefits for a small monthly fee. At a time when ISPs were introducing various blocks (mostly to stop people accessing torrent sites), such services had no problem finding customers. Also, as people grew more savvy to the dangers of using public Wi-Fi, and 'digital nomad' became a legitimate profession, their popularity soared.



NordVPN's Linux client is available in the Snap Store. It blocks ads, too.

Some of these VPN providers even had free offerings, and it's here that we find an early example of a VPN scandal. An outfit called Hola (which still exists; it's just a little more honest about how it operates now) offered a free browser extension that enabled users to connect to a VPN in the country of their choosing. The problem was that in order to subsidise this, users unwittingly became VPN end points. So, free users were effectively joining a peer-to-peer network in which their bandwidth was used, possibly to access dodgy material. Or at least material that other users saw fit to use a VPN to access. Hola also offered a paid-for service, Luminati, which used the same network but without enrolling users. In 2015, 8chan founder Fredrick Brennan announced that Luminati was being used, by way of the over nine million free Hola users, to perform DDoS (distributed denial of service) attacks on 8chan. Luminati, claimed Brennan, was a botnet for hire.

Hola denied this assessment, but was quick to update its terms and conditions to clarify that Luminati did indeed use free users' bandwidth. The fact that free users' bandwidth would be donated to other free users, it claimed, had always been in those Ts and Cs. It also admitted that a lone hacker had indeed used its service to perform the attacks. There are plenty of free VPN offerings today, and this episode speaks to the old adage 'there's no such thing as a free lunch'. Especially if 'lunch' is a metaphor for proprietary software that modifies network connectivity.

We'd like to say this kind of behaviour is rarer nowadays but, if anything, it is more widespread. In May 2024, the FBI dismantled what it claimed was "the largest botnet ever". Known as 911 S5, it was billed as a residential proxy service and, at the time of its demise, it had 19 million compromised IP addresses in nearly 200 countries. The botnet got so big by bundling free VPN applications (named *MaskVPN, DewVPN, PaladinVPN, ProxyGate, ShieldVPN* and *ShineVPN)* with pirated Windows software. Krebs on Security has an excellent write-up on this debacle at **https://krebsonsecurity.com/2024/05/is-your-computer-part-of-the-largest-botnet-ever/**.

Big-name VPNs are unlikely to indulge in this sort of malfeasance, but the sad truth is there's no real way of telling. Many of them have been audited by security professionals, who have verified operational details and logging policies. But nothing is stopping these

» CHEAP TRICKS

One unethical use for VPNs has been to take advantage of regional pricing on gaming platforms. Thanks to basic market forces, games are cheaper in countries with lower GDP. A fair few unscrupulous individuals have taken advantage of this by using a VPN in conjunction with a site such as **https://steamregionalprices.com** (definitely not an official Valve site) to determine where to get the best prices. Naturally, Valve takes a dim view of this, and now uses your billing address rather than your IP to get your location. We've heard you can get around this using gift cards (which are redeemable anywhere), but we don't expect this trick to last for ever.

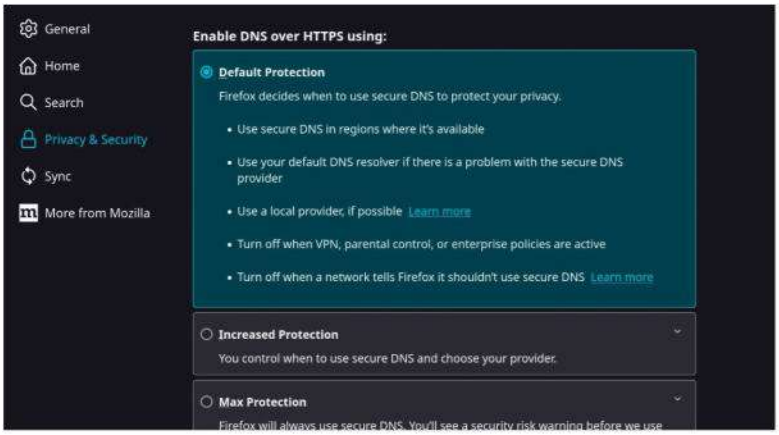
Another slightly dodgy use for VPNs is avoiding advertising on YouTube or other streaming platforms. At present, content is not monetised at all in Albania, Moldova, Myanmar or Ethiopia (and probably some others, too). So, if you can find a VPN provider with servers in those countries, you can dodge having your viewing interrupted by ad breaks. It's explicitly against YouTube's terms of service, though, so doing this while logged into your Google account might well win you a ban. Bans are also awarded to folk trying to subscribe to YouTube Premium from other countries.

outfits changing their configurations post-audit. It takes a brave soul to defy a law enforcement request, especially if it comes from any of the US's three-letter agencies. NordVPN once proudly claimed it took advantage of its Panamanian jurisdiction to ignore foreign law enforcement and government requests. But in January 2022, following the dismantling of VPNLab by Europol, it changed its tune. In its own words, it "will only comply with requests from foreign governments and law enforcement agencies if these requests are delivered according to laws and regulations". It also clarifies that its zero-log policy will be lifted (presumably on a per-user basis) if a court legally requires it to.

We should praise NordVPN's honesty here. Plenty of other outfits still claim they will happily flout law enforcement requests, but when the FBI comes knocking, we doubt they'd keep up that brazen attitude. NordVPN (and other more reputable providers) used to display a warrant canary on its website, which it said it would remove in the event of any government request. This year it has changed its policy to instead provide transparency reports (**https://nordvpn.com/blog/nordvpn-introduces-transparency-reports/**). At the time of writing, these show that it has received several million automated DMCA requests (likely aimed at people sharing copyrighted media over public BitTorrent trackers) and 81 enquiries from government institutions. But none of these has led to any user data being disclosed.

What use is a VPN?

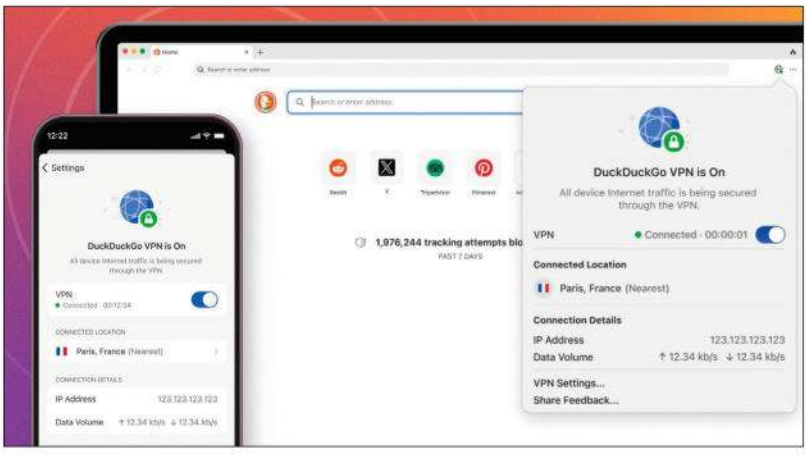
By hiding your IP, a VPN provides some degree of privacy. But an IP address does not necessarily identify you, and if it does, there are many other identifiers that sites or services can use to track you. Most obviously, if you sign in to your Facebook or Google accounts over a VPN, those companies know who you are. And if they have a list of known VPN IPs, they might know you're using a VPN, too. So, you'll probably have to complete an annoying captcha. The tracking cookies used by advertising firms are also unaffected by a VPN. Those firms might not know your name or email address (although Facebook and Google do), but they don't need to. They've already assigned you a unique tracking ID. Also, if you are unfortunate enough to have some sort of spyware on your machine, a VPN will not stop that phoning home and revealing your IP address (or doing whatever other bad things spyware does).



Firefox ships with DNS over HTTPS (DoH) enabled by default in some countries, so check your settings to see if it's on.

A VPN hides your traffic from your ISP. The other side of this coin is that it makes your traffic visible to your VPN provider. Since most websites use HTTPS (and other services use other encryption protocols), this doesn't mean they can see exactly what you're looking at, but it does mean they can see which IP addresses you're accessing. And from there, they can do reverse DNS queries to guess which sites you're looking at. Most VPNs provide their own DNS servers, too; unfortunately, many ISPs intercept DNS requests and force you to use their DNS servers. You can test for this using a number of websites (such as Surfshark's DNS leak tool at **https://surfshark.com/dns-leak-test**). A DNS leak means your ISP can see which domain names you are accessing; no DNS leak means your VPN provider can. If you trust a random VPN company over your ISP, perhaps you should think about changing ISP. DNS leaks can be mitigated at the browser level using DNS-over-HTTPS (DoH), but this brings its own problems (see **www.zdnet.com/article/dns-over-https-causes-more-problems-than-it-solves-experts-say/**)

Before Netflix's war on password sharing (from 2022 onwards), there was its war on VPNs. Content licensing hasn't quite caught up with the internet yet, so many shows are only available in certain countries. It was common to use a VPN to get around Netflix's geoblocking, allowing access to the latest US TV shows or obscure Japanese anime series. Around 2018 (when Netflix started producing a lot more content), VPN users found themselves facing an error message. Netflix had put in place detection mechanisms to stop users circumventing geoblocks. While some of these methods may have just been asking VPN providers nicely for a list of IPs they used, others were more sophisticated, using DNS leaks and WebRTC to detect anything suspicious. There are still VPNs that work with Netflix, some even charge extra for this. But don't count on it working indefinitely. Even if Netflix says it does hope one day to be able to license all content globally.



CREDIT: DuckDuckGo

Privacy-centric search engine DuckDuckGo offers a VPN and personal information removal service, too.

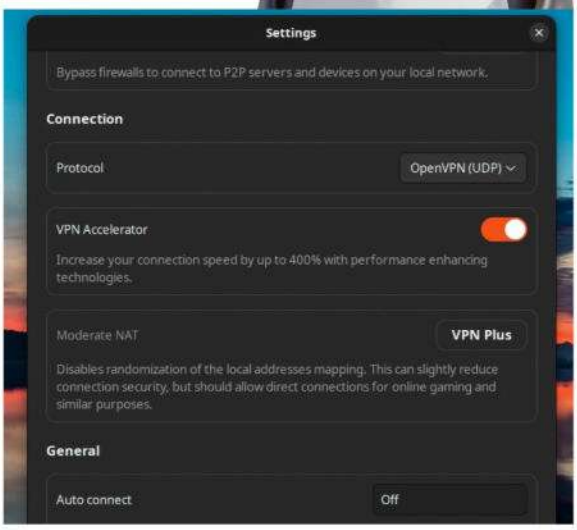
Get started with Proton VPN

It's easy, fast and free (or you can pay if you want). Proton VPN is one way to see what a VPN can do for you.

Based on the previous two pages, you might get the impression that most VPNs are operations run or used by ne'er-do-wells. That might be true, but we didn't say it. You might well be looking for advice on how to run a VPN on Linux. And if that's the case, we can help. A number of VPN providers boast Linux support, but on the flip side, we've seen anecdotal evidence that this support may not really be all that good. We won't name any names here, but just because a VPN offers a Linux client, it doesn't mean your experience using it will be smooth. You also should, as we've hinted earlier, be very cautious about running proprietary VPN tools on your machine.

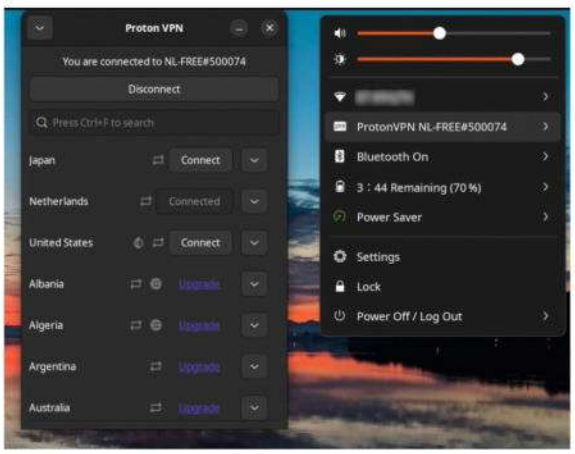
Ideally, you want your VPN provider to integrate directly with *NetworkManager* (or whatever you use to administer your connectivity). This way, you won't run into any difficulties running its own client (which may only work on a particular version of a particular distro, and may even not work particularly well there). If you're only interested in routing web traffic through a VPN, most providers maintain a *Chrome* or *Firefox* plugin. If you look on the Chrome Web Store or Firefox Extension pages (or the equivalent for your favourite browser), you'll find no shortage of free VPN tools that purport to offer maximum privacy and security. Again, be wary of these. Some of them boast hundreds of thousands of users, but that doesn't mean you should trust them.

If you want a free VPN, one company you can probably trust is Proton, the privacy-focussed Swiss company behind the popular ProtonMail service. Its Proton VPN offering launched in 2017 and has over 5,500 servers in 91 countries. Don't get too excited, though; the free tier currently only allows you to choose from six countries (this depends on your region – in our tests, only Japan, the Netherlands and US were available). We should also stress that Proton did not sponsor this feature in any way. We just quite like its service and the fact that it has open sourced all its code. In the interests of equality, we'll also give a nod to Mozilla and Mullvad, both of which are premium services that have open source clients. Windscribe has a free tier and also has open source clients. PureVPN, ExpressVPN and Private Internet Access (PIA) all support Linux.



You need to pay to unlock some settings, but you do get the glorious-sounding VPN Accelerator for free.

Proton's VPN client is simple and works well on most Gnome-based desktops. It integrates with NetworkManager as well.



ExpressVPN open sourced its Lightway protocol in 2022, but in a world where WireGuard is becoming the norm *(see over)*, this hasn't seen much adoption. We should also note that ExpressVPN's former chief information officer was fined $335,000 by the US government in December 2023. It was revealed that he was part of Project Raven, an outfit working with the United Arab Emirates to spy on human rights activists, journalists and rival governments (see **https://cybernews.com/news/expressvpn-cio-daniel-gericke-fined-335-000-for-cyber-espionage/**). There's no evidence the CIO conducted espionage on ExpressVPN users, but it's still not a good look. ExpressVPN, PIA and Cyberghost are all owned by the UK-based Kape Technologies, so it's unclear (we're not lawyers) which laws have jurisdiction over any potential surveillance.

Installing the Proton VPN app is easy – officially it supports Debian, Ubuntu and Fedora, but it should work on any Gnome-based desktop. We had no problems

running it on Pop!_OS. The official instructions are at **https://protonvpn.com/support/linux-vpn-setup/** if you run into any difficulties. An official Flatpak is available on Flathub, too, if you'd rather use that (on Steam Deck, for instance). We'll show you how to install it via its own DEB package on Ubuntu. Open a terminal and type:

```
$ wget https://repo.protonvpn.com/debian/dists/stable/main/binary-all/protonvpn-stable-release_1.0.3-3_all.deb
$ sudo dpkg -i ./protonvpn-stable-release_1.0.3-3_all.deb && sudo apt update
```

That installs the entry for Proton VPN's repo and updates the package cache. To actually install it, use:

```
$ sudo apt install proton-vpn-gnome-desktop
```

Before you start it, you need to sign up for the service. Go to **https://protonvpn.com/pricing** and scroll down to the Proton Free section. You just need to give an email address and choose a password. Note that the free service only supports one device at a time. The paid-for service (VPN Plus) currently costs €4.49 per month if you sign up for two years, which is pretty competitive. The app should be available from the applications grid (press the Windows key and type the first few letters of `proton vpn`, but if not, you can start it from the terminal with `protonvpn-app`. Now choose a country and the app deftly sorts out the connection. Different servers support different features, such as peer-to-peer file-sharing, Tor and smart routing (so you can use app that usually rely on port forwarding).

Depending on your version of Ubuntu (or what else you have installed), you may see a status icon in the top-right. If not, see the official instructions for how to add system tray icons. Regardless, once connected, you should also see a new icon in the top-right indicating a VPN connection, due to Proton VPN's integration with *NetworkManager*. If you don't want to use the app (or it doesn't work for you), it's possible to set up a connection manually – just check the official docs.

One of the app's main features (and one that is expected of most consumer VPNs) is a kill switch. We're all used to our internet connections dropping from time to time, whether it's due to poor Wi-Fi or unreliable ISPs. But when the connection comes back, it generally does so without restoring the VPN link. If you're a journalist in a repressive regime, this could spell disaster, since your browser might automatically reconnect to whatever websites you were viewing. A kill switch averts this disaster by shutting down all external network access until the VPN is reconnected. Obviously, this is annoying if you're only using a VPN to access certain services (we don't judge), so this is off by default. To enable Proton VPN's kill switch, open the drop-down menu in the top-left of the app and choose Settings. It's the third option down. You can read more about how it works at **https://protonvpn.com/support/what-is-kill-switch/**.

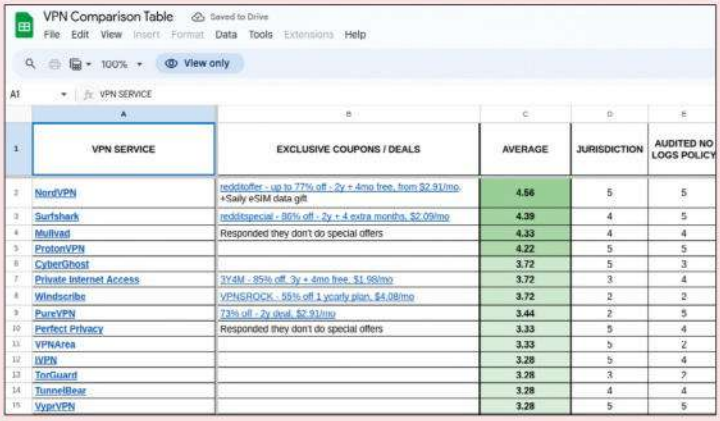
A number of Proton VPN's options are only available to paying customers, but its VPN Accelerator is enabled by default. This, as far as we can tell, is a marketing term to describe multithreading, smart traffic routing and kernel tweaks on Proton's infrastructure to increase speed and stability. More information is available on its website. Try turning it off if you run into difficulties with whatever purposes you're using its service for. Other VPNs have similarly named features, sometimes billed as a way to get around traffic shaping by your ISP (if it limits the speed of VPN connections). You can also set Proton VPN to auto-connect to a particular server when it starts, and then if you set the app itself to auto-start (in Ubuntu's *Startup Applications* tool), you can have a VPN connection from the get-go. Proton VPN uses the venerable OpenVPN protocol under the hood, by default using UDP connections. You shouldn't need to change this, but the application lets you switch this to TCP if you feel the need.

VPN COMPARISONS

If you want more comprehensive advice about which VPN company to support with your wallet, there are a few options. The /r/VPN subreddit maintains a colour-coded spreadsheet, which rates VPNs on policy, jurisdiction, encryption and a whole bunch of other metrics. You can find it at **www.reddit.com/r/VPN/comments/m736zt/vpn_comparison_table/**.

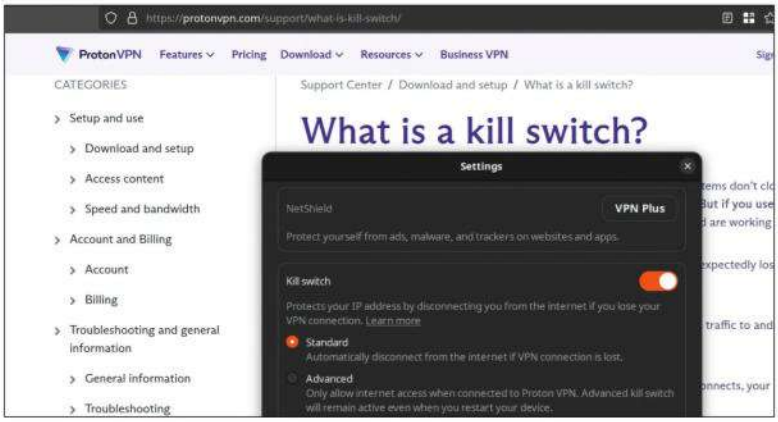
One of the more interesting columns of that table is the RAM Servers one. These have been en vogue for the past couple of years. Sometimes called diskless servers, they refer to machines that boot via PXE from a network image and do everything in memory. This adds credence to any no-log policy, since in the event a RAM server (or diskless node, call it what you will) is hacked and has any log file settings changed from **/dev/null** to any local filesystem location, that location still behaves like a black hole.

And seeing as we're singing Proton's praises, here seems like a good place to mention it's just launched a new password manager. *Proton Pass* is open source (of course) and integrates with the PAM (Pluggable Authentication Module) subsystem, so your passwords can be unlocked when you log in (if you want).



Everyone at Future Towers loves a giant Google Sheet.

It is more or less de rigueur for commercial VPNs to offer a kill switch capability.



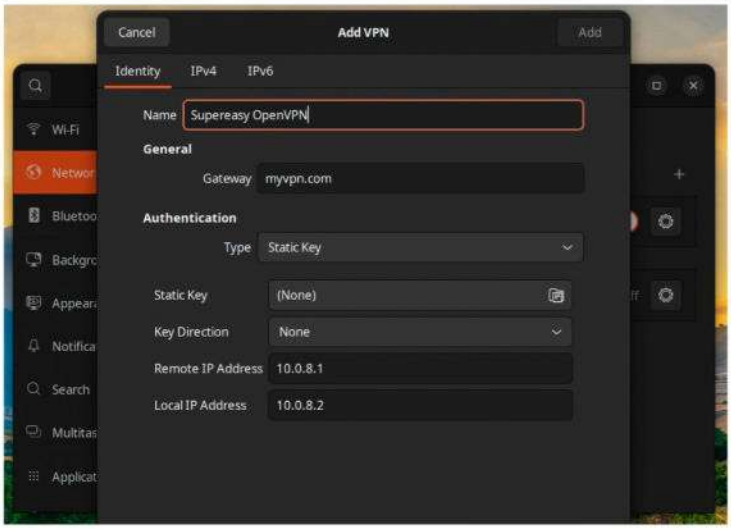
Running your own VPN

If you want a job done properly... We show you how to host your own VPN, at home or on a VPS, with OpenVPN or WireGuard.

If you think you can do better than some of these fly-by-night VPN providers, it's fairly simple to set up your own VPN. You can do this on your home machine, so that when you are out and about, whatever networks you're connecting over can only see that you are connecting to your ISP. Your ISP, of course, can see what you're up to exactly as though you were working (or playing) from home. See the box *(below)* for guidance on how to host a remote VPN server.

Let's start with a (very) basic overview of setting up an OpenVPN server. We won't cover initial setup and securing of the box (be it real or virtual), otherwise we'd be here all day. OpenVPN is a hugely complex and powerful bit of software, but in its simplest configuration (with a single static key), it's amazingly straightforward to fire up. First we need to generate a key, which you may as well do on your local machine:

```
$ openssh --genkey --secret static.key
```



You can add your OpenVPN connection into NetworkManager for easy one-click (OK, maybe two) access.

HOSTING YOUR OWN VPN

The most common way to run a VPN is using a cheap VPS (virtual private server). If you already have a server (virtual or actual), you can use that, but in the interests of security, it's best to keep things separated. Cheap VPSes can be had from the likes of Digital Ocean or Mythic Beasts, and you can generally go for the lowest spec offered – a VPS doesn't require much in the way of compute or memory. At least if it's just for personal use.

Paranoid individuals might want to rent a VPS in a country outside of the Five Eyes alliance (Australia, Canada, New Zealand, UK and US), since those countries will happily help each other spy on your server. Remember, the encryption is only between you and said server – anything after that is as vulnerable to snooping as if you were not using a VPN. Amazon Web Services (AWS) might well offer cheap Elastic Cloud Compute (EC2) instances in diverse regions, but it's still an American company and is unlikely to resist, say, Foreign Intelligence Surveillance Act (FISA) warrants from the US government. Not that you would be doing anything that would interest them, of course.

As the second parameter indicates, this is a secret key, so be careful where you store it (don't put it on your GitHubs or Nextclouds). The key does need to be on your server as well (which might be your local machine if you are wanting to run the OpenVPN server from home). Copy it to the server (which we've imaginatively called **yourserver.com** with username user) with:

```
$ rsync static.key user@yourserver.com:/home/user
```

We need to install OpenVPN on both the client and the server:

```
$ sudo apt install openvpn
```

First we'll configure things server-side, so log in to the remote machine and create a configuration file with (if you're not using Ubuntu, omit the **server/** subdirectory in the path below):

```
$ sudo nano /etc/openvpn/server/server.conf
```

And populate it with our basic initial configuration:

```
dev tun
ifconfig 10.8.0.1 10.8.0.2
secret /home/user/static.key
```

You can use any private IP addresses you like in the second line (for example, anything beginning with **10.*** – don't use your real IPs or it will break). The first is the server end point and the second is that of the client. When it's activated, both gain a new virtual network interface called **tun0** (short for tunnel). Start the server listening on the remote machine (it doesn't need root privileges):

```
$ openvpn --config /etc/openvpn/server/server.conf
```

If you see any messages, refer to the official documentation at **https://openvpn.net/community-resources/static-key-mini-howto/**. There are also some useful additional options on this page (for

example, if you want to run it as a daemon or use compression). Now we'll make a client configuration on the local machine:

```
$ sudo nano /etc/openvpn/client/client.conf
```

Again, we're assuming you're using Ubuntu here, and that you created the key in your **home** directory. Add the following content to the file:

```
remote yourserver.com
dev tun
ifconfig 10.8.0.2 10.8.0.1
secret /home/user/static.key
```

The IPs should be the same as in the server config, albeit transposed. Amazingly, that's all that's required for our OpenVPN setup. Let's try it out with:

```
$ openvpn --config /etc/openvpn/client/client.conf
```

Barring any error messages, you should be able to ping the **10.8.0.1** address from your local machine. OpenVPN takes care of all the routing, so all data (except LAN traffic) flows through the **tun0** device. Great victory!

We should note that the above configuration is fairly rudimentary. If you want anyone other than yourself to be able to access the VPN, you need to share the key with them. And if they have the key, they can (with some effort) spy on your connection. More users also increases the risk of the key being compromised. So, the more grown-up way to run OpenVPN is with PKI (public key infrastructure). This way, everyone can have their own keys and security is embiggened. We'll leave further explanation to the excellent DigitalOcean tutorial at **www.digitalocean.com/community/tutorials/how-to-set-up-an-openvpn-server-on-debian-11**.

En (Wire)Garde

Because OpenVPN is much more complicated than we've made it seem here, there has long been a need for a simpler, more robust and potentially faster solution. In 2018, Jason A Donenfield officially released such a remedy in the form of WireGuard. It has since gained much traction and has been included in the Linux kernel since version 5.6 (released in 2020). So, it'd be remiss of us not to finish up with a quick guide to it. As before, we start by installing the WireGuard userspace on both client and server:

```
$ sudo apt install wireguard
```

First we need a private/public key pair on the server. WireGuard is rightfully fussy that the private key is only accessible by the user who created it. So, we'll set this up with root as the owner of the key file (since that's who will initially start the daemon):

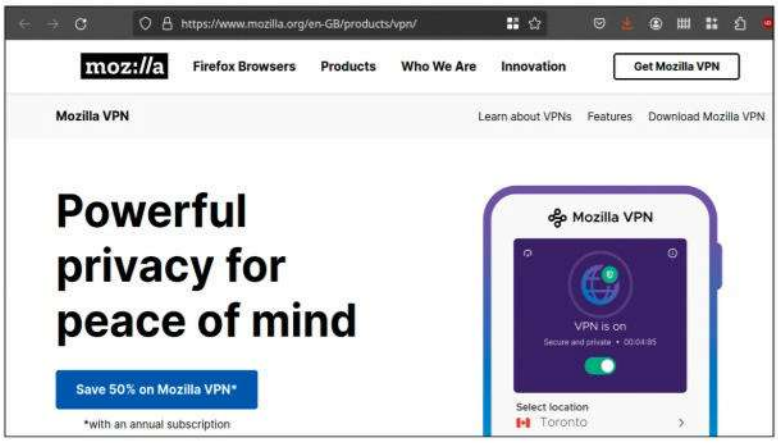
```
$ wg genkey | sudo tee /etc/wireguard/private.key
$ sudo chmod go= /etc/wireguard/private.key
```

The base64 encoded private key will be outputted. Copy it to the clipboard to save yourself some time in a moment. Now generate a corresponding public key:

```
$ sudo cat /etc/wireguard/private.key | wg pubkey | sudo tee /etc/wireguard/public.key
```

Now we set up the WireGuard server config. First create it with `sudo nano /etc/wireguard/wg0.conf`, then populate it with the following:

```
[Interface]
PrivateKey = <paste private key here>
Address = 10.8.0.1/24
ListenPort = 51820
SaveConfig = true
```



Mozilla's VPN offering uses WireGuard under the hood. It has been fighting for privacy since 1998.

We need to configure the server to forward and relay VPN traffic, too. Add the following line to **/etc/sysctl.conf** (as root):

```
net.ipv4.ip_forward=1
```

Now run `sudo sysctl -p` to activate it. See the docs for IPv6 configuration, and also how to configure any firewall you may be running. Now run the following:

```
$ iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o eth0 -j MASQUERADE
```

Now we're ready to configure our client. Before we do, copy the public key we generated earlier. On the local machine, run the same three commands from earlier to generate a key pair. Then create the peer configuration file at **/etc/wireguard/wg0** with the following contents:

```
[Interface]
PrivateKey = <client's private key>
Address = 10.8.0.2/24
```

DO IT YOURSELF

"When you are out and about, whatever networks you're connecting over can only see that you are connecting to your ISP."

```
[Peer]
PublicKey = <paste public key from server>
AllowedIPs = 10.8.0.0/24
Endpoint = <server's IP address>:51820
```

Now copy the public key we generated on the local machine (honestly, this rigmarole is very nearly complete). And back on the server, open up the configuration and add this stanza:

```
[Peer]
PublicKey = <paste public key from local machine>
AllowedIPs = 10.8.0.0/24
```

Now we can start the server with `wg-quick up wg0`. Then run the same thing on the client and you should be good to go. For a more in-depth look at WireGuard, see the excellent tutorial at **https://dev.to/tangramvision/what-they-don-t-tell-you-about-setting-up-a-wireguard-vpn-1h2g**. LXF

The best VPNs

We asked **Andreas Theodorou**, VPN chief at TechRadar Pro, to reveal his top picks for the fastest and most secure VPNs on the planet.

HOW WE TESTED...

VPNs are notoriously difficult to test. Their performance can vary a lot across different global locations and device types – and figures from speed tests and other benchmarks tend to change over time, so you need to retest regularly. Plus, VPN services are by nature secretive black boxes, so it's hard to verify claims such as a no-logging policy. The process is to subject these VPN services to a custom array of tests designed to determine which ones to avoid.

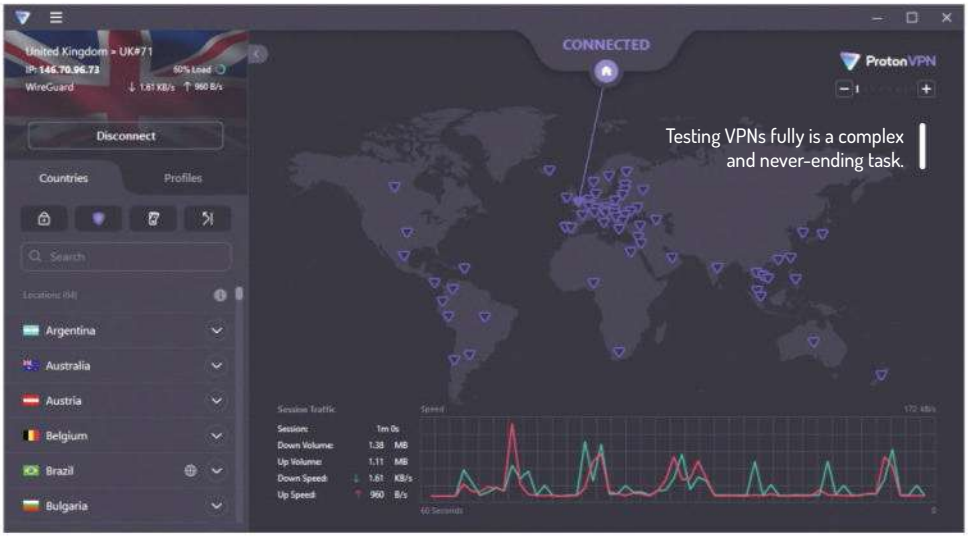
Features We gather as many technical details as possible, analysing things such as network size, server locations, and supported protocols and encryption methods.

Security We analyse and probe privacy policies and encryption methods, and pay attention to any independent security audits that the provider has undergone. We also run our own tests to verify DNS leak protection and kill switches.

Performance Using multiple device types over a 1Gb/s internet connection, we measure connection speeds multiple times at regular points during the course of a day (and regularly throughout the year), and check connection times.

Unblocking abilities We test the ability to unblock region-restricted content across global platforms such as Hulu, Netflix, Disney+ and more.

Support We engage with the customer support offerings, such as live chat and email. We also analyse the help sections on the providers' own websites.



What's that little URL that lives at the bottom of the pages of *Linux Format*? It's our sister website TechRadar Pro. Those lot have the resources and person-power to continually test and update reviews of all sorts of professional online services, including VPNs. It has zillions of in-depth VPN reviews testing speed, streaming, support, privacy and value. So, to help you get started choosing the best VPN for you, we've got TechRadar Pro to suggest six of the best. As technology never stands still, we recommend you keep tabs on **www.techradar.com/vpn/best-vpn** for updates on who's offering the best services.

Make sure you choose the right VPN for your needs. We can't tell you which one that is – we can only show you the cream of the crop to help you make an informed decision. But if you're not quite sure what to look for, here are the most important considerations:

Price Arguably the most important factor – make sure it's within budget. Whether you get a premium, cheap or free VPN, just make sure it's safe.

Apps Does it support your device? If so, are the apps easy to use, or are they confusing and cluttered? The less you know about VPN tech, the simpler you want to go.

Features Does it have all the usual mod cons you'd expect from a VPN – a kill switch, auto-connect, split tunnelling? If not, you may want to look elsewhere. What other value propositions does it have? Maybe you want a built-in password manager or antivirus.

Security Is it meeting industry standards for encryption with WireGuard and OpenVPN? Is it offering quantum-secure key exchanges to protect you from future data breaches?

Privacy Does it have a no logs policy, or just say it does? Some providers claim there's no logs, but in reality, they're just not logging a specific thing – and fleecing you elsewhere. Remember, if there is even a shred of data being collected that could link back to you, you don't want to use that service.

Unblocking While most, if not all, VPNs claim to unblock everything everywhere, only a few can actually put their money where their mouth is. If unblocking content is your priority, get a VPN that's proven to work.

Performance It's not just about download speeds – you want consistent and reliable connections that don't randomly drop or cause immense packet loss.

Locations Are there servers in your area (or the area you want to unblock)? If so, are they physical or virtual servers? Virtual servers tend to be a bit of a gimmick, and often become slow and congested, so if you see a provider saying they have tens of thousands of servers to pick from, just be wary. Remember, it's about quality, not quantity.

Support Is there live chat support or is it just an email ticketing system? We've seen it all, and the better the customer support, the more invested the VPN is in giving you a better experience. Look for good knowledge hubs filled with helpful articles and a diverse range of contact methods.

CREDIT: ProtonVPN

Private Internet Access

It can be difficult to get a VPN client set up on Linux. The process often requires confidence with the command-line interface. This is why PIA is our top pick for Linux users. With an easy installation process and a fully featured GUI app, PIA makes using a VPN on Linux a breeze.

PIA is also excellent for power users who need lots of flexibility regarding installation and configuration. Getting into the settings can be a bit overwhelming for new users – there's a huge number of features on offer – but if you don't need all that additional functionality, there's a nice, simple connect button to get you online.

There's support for an unlimited number of devices, split tunnelling to allow you to choose which apps use the VPN and which don't, and PIA Mace which is an ad/tracker/malware-blocking tool. There's also multi-hop connections for additional security, obfuscated servers to hide your VPN usage completely, support for port-forwarding to optimise torrent downloads, and more.

PIA is weak when it comes to speed and unblocking ability. It struggled in our tests, with WireGuard speeds dropping to 330Mb/s and OpenVPN down to 270Mb/s.

This is still more than enough for multiplayer games and 4K streaming, so while we would hope for better results, PIA remains very usable. You can still take advantage of its ability to unblock popular streaming sites, including Netflix, Prime Video, Disney+ and more.

While PIA's speeds are nothing to write home about, it still shines as the best VPN for Linux users and has gone to great lengths to ensure that. PIA also has an impressive set of tools and features, including a GUI client for less experienced users, coupled with strong unblocking capabilities. This is a great option for Linux users, as well as for anyone who's already signed up and happy with the service. ***Andreas Theodorou***

We found the PIA Linux client and support to be top-notch.

VERDICT

DEVELOPER: PIA **WEB:** www.privateinternetaccess.com **PRICE:** £10.99pm (£1.57pm 2-year)

» Rating 9/10

CREDIT: PIA

NordVPN

Nord offers an unmatched blend of features, speed, unblocking and value. Not only is it half the price of ExpressVPN, it also has more features compared to Surfshark, its two nearest rivals.

NordVPN offers all the industry-standard features you'd expect. Log into the app, select a location, and you can connect to servers all over the world instantly – encrypting all your web traffic and changing your IP address in the simple click of a button. It's easy to use, even for beginners, and has all the advanced features you'd expect if you're more experienced.

With a kill switch and an auto-connect function, you automatically connect to a VPN server when you boot it up, and if the connection drops, Nord automatically quarantines your internet traffic. That's essential if you don't want your location or identity leaking by accident. You can even choose what you don't want going through the VPN, thanks to split tunnelling.

But there's so much more than just a fast and reliable connection – NordVPN sports an impressive suite of additional privacy, security and usability features. There's Threat Protection, which protects you by blocking malicious sites and scanning downloads for malware. Threat Protection has now been split into Pro and Lite, with Pro offering even more features that go well beyond the standard DNS-based blocking. However, while this is a great tool, if you get a virus, you need a dedicated antivirus app to remove it. Nord's core DNS-based blocker is preventative, not curative.

Alongside this is NordVPN's dark web monitoring, which scans the dark web for your credentials and email address to ensure they haven't been compromised. There are obfuscated servers that can disguise your VPN usage, and a feature called SmartPlay to help you access streaming content when not in your home country. ***Andreas Theodorou***

Widely known for its excellent VPN, it seems its reputation is well deserved.

VERDICT

DEVELOPER: NordVPN **WEB:** https://nordvpn.com **PRICE:** £10.39pm (£2.79pm 2-year)

» Rating 9/10

»

CREDIT: NordVPN

ExpressVPN

A significantly higher monthly subscription price and a more limited feature set compared to NordVPN hold ExpressVPN back, but it's got plenty to offer anyone in the market for a VPN.

It's a strong choice if you're concerned about keeping your data safe online, with a widespread selection of servers to pick from, robust security tools and protocols, and excellent speed results. But the main reason we love it so much is because it has the most user-friendly apps of any other service, so it's a brilliant option if you're new to VPNs.

For the more security-minded user, its hard to argue with ExpressVPN's offering. Not only does Express have its own bespoke VPN protocol in Lightway, it also offers all the main features provided by Nord, including auto-connect and a safety-net-providing kill switch.

ExpressVPN calls its kill switch tech Network Lock. It ensures you can only access the internet via the VPN's secure connection, and blocks all traffic should the connection drop, preventing data being leaked.

It comes with a built in ad-blocker, plus an optional password manager called ExpressVPN Keys that you can use if you don't already have one. These days, when everyone has a hundred different accounts, a password manager is an essential tool – making it easier to log into each different site, while keeping your details secret and safe at the same time.

CREDIT: ExpressVPN

With the easiest-to-use apps and a hyper-focus on user privacy and security, we think ExpressVPN is the best choice for anyone primarily looking for a privacy-focused VPN service, and who is happy to pay a little extra. It's also an ideal VPN if you're a beginner and just want things made that extra 10% easier. For everyone else, you might be better off choosing either Nord or Surfshark, both of which cost less than half the price of ExpressVPN. ***Andreas Theodorou***

You have to pay for excellent service and that works for ExpressVPN.

VERDICT

DEVELOPER: ExpressVPN **WEB:** www.expressvpn.com **PRICE:** £10.55pm (£6.78pm 1-year)

» Rating 9/10

Surfshark

Surfshark isn't just our pick as the best cheap VPN, it's also a one-stop security suite. It has built-in ad-blocking, antivirus, tracker-free web browsing, and a data breach notifier built-in – not to mention a tool called Alternative ID, which generates a brand new identity and email address to use online to protect you from spam or having your details harvested. It's premium protection without the price.

One of the crown jewels of Surfshark's features list is Incogni, a service that automates the process of requesting the deletion of personal data stored by data brokers and advertising companies. This gives users a chance to push back and reclaim at least some of their digital privacy from invasive data-tracking practices.

Surfshark also offers other security features, such as Camouflage Mode, its implementation of obfuscated servers, and CleanWeb, its ad/tracker/malware-blocking solution. It also offers unlimited simultaneous connections for busy households with lots of devices.

Surfshark is the king of the hill when it comes to our speed tests, reaching speeds of over 950Mb/s when connected through WireGuard, and a chart-topping 640Mb/s using OpenVPN. That's why it's our top pick as the best super-fast VPN currently available.

We also found it can unblock multiple Netflix libraries as well as Disney+, BBC iPlayer and most others, making it a solid and cost-effective choice for those who want to stream foreign libraries. The only real downside is if you live in China – Surfshark struggles to reliably defeat China's Great Firewall and there are better options available.

CREDIT: Surfshark

Super-speedy and pretty affordable makes it a snappy choice.

Surfshark has a low monthly subscription cost, super-fast download speeds, and an impressive array of privacy tools – a budget price doesn't have to mean budget features. Most people will find that Surfshark fully meets their expectations. ***Andreas Theodorou***

VERDICT

DEVELOPER: Surfshark **WEB:** https://surfshark.com **PRICE:** £12.29pm (£1.69pm 2-year)

» Rating 9/10

ProtonVPN

All VPN providers claim to be security experts, but few can match Swiss-based Proton's experience and track record. The company has run popular end-to-end encrypted email service Proton Mail since 2014, and its other products include a secure calendar and encrypted cloud storage.

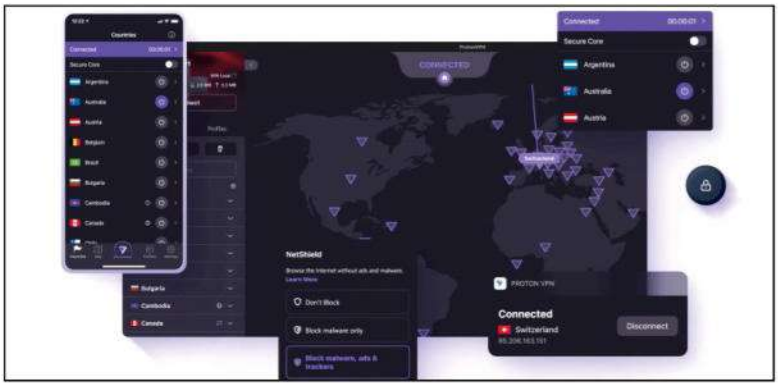
A lengthy feature list ensures the service scores just about everywhere. Proton VPN is P2P-friendly, supports up to 10 simultaneous connections, has a kill switch, DNS leak protection, and built-in Tor support for accessing Onion sites. A versatile split-tunnelling system allows you to route specific app or destination IP traffic outside of the VPN, and WireGuard support aims to get you the best possible performance.

ProtonVPN's free service has dropped its ability to choose locations, so now all you can do is hit a Fastest button and the app chooses the best server from five countries: Japan, Netherlands, Poland, Romania, USA.

That makes the service less useful for unblocking, and could introduce new problems. Previously, if you chose a location, you knew that's what you'd get. Now, if you're a similar distance from two or three locations, you might get a different one each time. That could be bad news if you're visiting websites that change how they work depending on your location.

WireGuard speeds got an incredible boost this time around, peaking at 950Mb/s. That positions Proton VPN among the speed leaders. If you can't use WireGuard for some reason, Proton VPN also supports the industry standard OpenVPN protocol. We got some great performance here, too, reaching 400Mb/s.

Proton VPN unblocked almost everything we tried, its powerful apps are open source and independently audited, WireGuard speeds can be excellent, and there's a free plan with no bandwidth limits. This is a great VPN, and it's getting better. ***Mike Williams***



CREDIT: ProtonVPN

There is an excellent free offering, which is why we've featured it.

VERDICT

DEVELOPER: ProtonVPN **WEB:** https://protonvpn.com **PRICE:** $9.99pm ($4.49pm 2-year)

» Rating 8/10

Windscribe

This is an interesting VPN with great value commercial products and loads of features, yet is easy to use, with a generous free plan.

An array of apps keeps you covered on Linux, Windows, Mac, Android and iOS. *Chrome, Firefox* and *Edge* extensions give you even more ways to connect, and the website has guides to help you set up the service on routers, *Kodi,* Amazon Fire TV, *Nvidia Shield,* and via any OpenVPN-compatible software or device.

WireGuard, IKEv2 and OpenVPN support with strong AES-256 encryption keep all your VPN tunnel traffic safe from snoopers, while stealth technologies try to obfuscate your VPN usage, perhaps allowing you to get online even in countries that block VPN traffic.

Its apps look great and are easy to use for beginners but also include many advanced features, including split tunnelling, MAC address spoofing (a clever way to reduce the chance of being tracked), versatile auto-connect rules, full IPv6 support, and even a command-line interface to automate the VPN from scripts.

ROBERT is Windscribe's DNS-based tool for blocking ads, malware, trackers and online content such as gambling, porn, fake news, clickbait and so on.

There's no 24/7 support but Windscribe does have a decent web knowledge base and a helpful support chatbot. You can raise a ticket if you need more advice, and in our experience, replies are detailed and helpful.

Windscribe's free plan offers a generous 10GB of data transfer a month if you register with your email address, and 2GB if you don't.

Windscribe is a likable VPN, with good-looking and powerful apps, expert-level features, and one of the most generous free VPN plans around. The array of advanced options and settings means this probably isn't the best choice for newcomers or anyone only looking for the basics. If you're unsure, there's an easy and risk-free way to decide: install the free version and see how it works for you. ***Mike Williams*** LXF

CREDIT: Winscribe

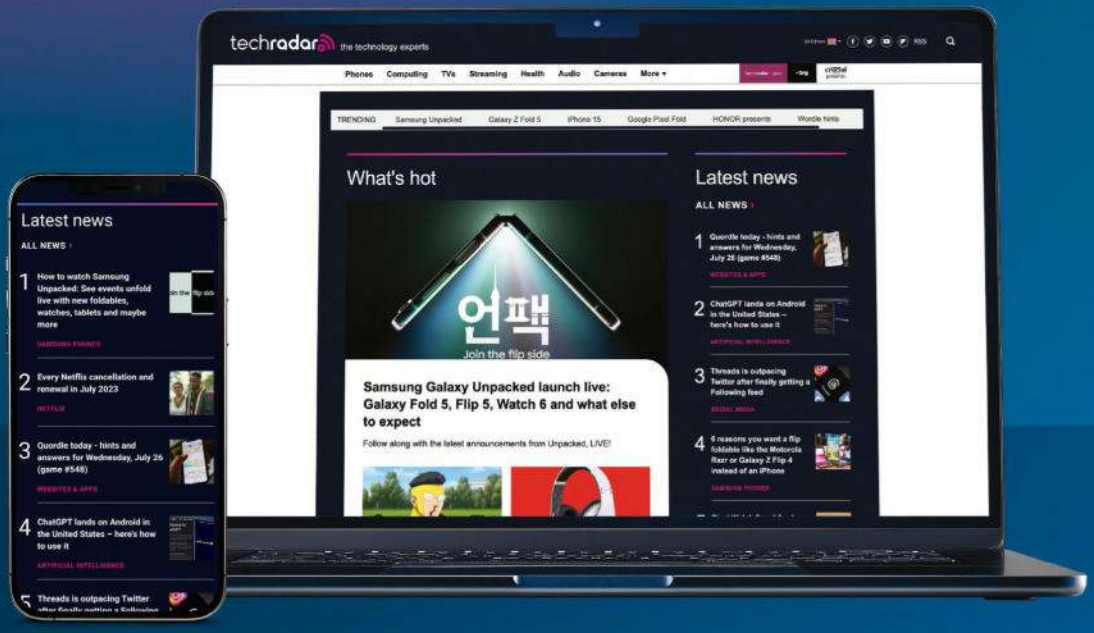
A Linux GUI app is available and there's solid support, with free options, too.

VERDICT

DEVELOPER: Windscribe **WEB:** https://windscribe.com **PRICE:** $9pm ($5.75pm 1-year)

» Rating 8/10

Meet the technology experts



- The world's **most** comprehensive technology website
- An **unrivalled** mix of news, opinions, reviews and features
- **All-new design**, new homepage, new features and special reports
- Backed by **over 300 years** of editorial experience

techradar
the technology experts

www.techradar.com

Potential speed boost for the Pi 5

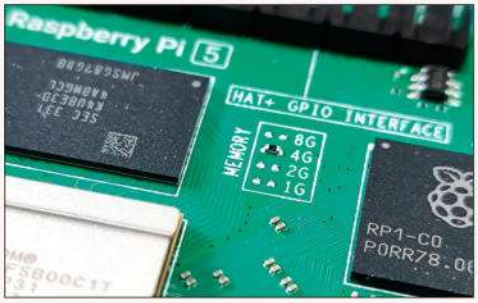
New memory access schema could deliver significant single-core and multi-core boost.

Igalia, the free software consultancy known for its work on the Raspberry Pi's GPU, has revealed that it is investigating NUMA (non-uniform memory access) emulation for ARM64 devices. The investigations have so far yielded a potential and significant performance uplift for the Raspberry Pi 5, discussed on a Linux kernel list via a message from Tvrtko Ursulin.

According to the post, "This series adds a very simple NUMA emulation implementation and enables selecting it on ARM64 platforms." Figures determined using *Geekbench 6* test runs reveal that this improves single-core performance by 6% and multi-core performance by approximately 18%.

Ursulin explained the concept in a little more depth: "[S]plitting the physical RAM into chunks and utilizing an allocation policy such as interleaving can enable the BCM2721 memory controller to better utilize parallelism in physical memory chip organization."

NUMA allows each CPU to have its own bank of locally attached memory while still having access to memory connected to other processors in the system. This results in fast latency for 'near' memory (locally attached) but slightly slower latency for 'far' memory – memory attached to the other processors.



New memory access could bring bonus speed.

A class being taught Scratch.

CREDIT: Raspberry Pi Foundation, Lifelong Kindergarten

Odisha calling

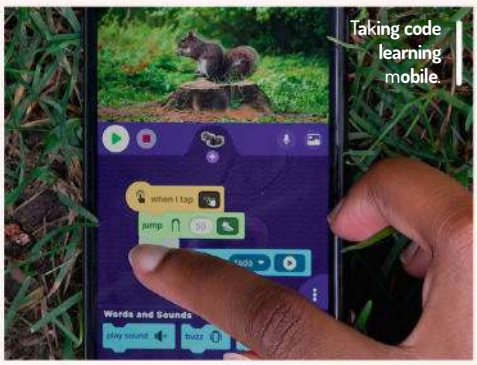
Expanding learning.

Code Club has already landed in Odisha, India, and now the aim is to expand this success with a full high school curriculum and resources for supporting teachers. Last year, 311 'master' teachers were trained with the aim of expanding training to 8,000 schools. Find out more on this ambitious project: **https://bit.ly/lxf318odisha**

Dr OctoStudio

Tentacles get into code.

Scratch is an amazing learning resource but it struggles on mobile devices. So, say hello to *OctoStudio!* Designed on the back of requests from educators, it's aimed at phones and tablets, where children don't have access to laptops or desktop computers, but do have access to mobile devices. Learn more: **https://octostudio.org**



Les Pounder works with groups such as the Raspberry Pi Foundation to help boost people's maker skills.

» BUY, BUY RASPBERRY PI

The recent Raspberry Pi initial public offering (IPO) saw the company behind our favourite single-board computer offer its shares for sale on the London Stock Exchange. Yes, now anyone can buy shares in the company and support the project's future.

On 11th June, shares started trading at 280p each. This quickly rose to 354p by mid-morning. The share price peaked on 17th June at 440p per share, dropping to 372p by 20th June. After a day of stagnation, the share price rose to 402p by 27th June, ending trading on 28th June at 400.50p. What does this all mean? Basically that shares are selling for more than their initial estimate, and Raspberry Pi is set to make plenty of money to fund its future. The money from the IPO will be used to invest in equipment and processes that will shape the future of the company and its products. How much input shareholders will have on this remains to be seen.

The IPO marks a transition for Raspberry Pi, from a company with 10,000 Raspberry Pis in the late Jack Lang's garage, to the 60 million units sold across the world, and onwards to a hopeful future with even more Raspberry Pi.

You can track the progress of Raspberry Pi's shares via the London Stock Exchange website: **www.londonstockexchange.com/stock/RPI/raspberry-pi-holdings-plc/company-page**.

Whether or not you make any money on the purchase is largely in the hands of the ever-volatile financial market, so do your research and seek advice from experts before cashing in your pension to purchase shares.

FydeOS

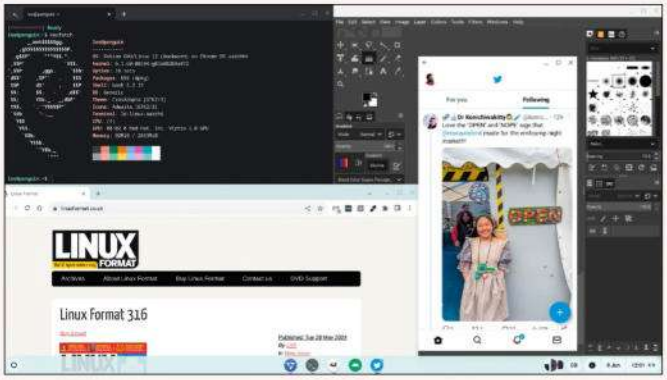
Les Pounder loves his Chromebook and his Raspberry Pi 5 – wouldn't it be lovely if the two could be combined?

IN BRIEF

A ChromiumOS alternative to Google's ChromeOS, FydeOS provides largely the same experience but without the need to buy into Google's hardware and software ecosystem. Available for the Raspberry 4, 400 and 5, this is a great OS choice for those who don't need all the power of a full OS.

Chromebooks are damn handy. We've got one for when we need just enough power for answering emails, browsing and YouTube video playback when we are on the move. ChromeOS is not available for every device, but FydeOS is, and it is pretty much the same thing. We took the latest version for a spin on a Raspberry Pi 5 with 4GB of RAM.

Installation is simple. Download the disk image, fire up *Raspberry Pi Imager* or *Balena Etcher,* and insert a 16GB or larger microSD card. Eject the card, put it in your Pi 5 and power up. We then encountered the post-installation setup steps. Language selection is the first step, and then we're asked if we want to set up or use a FydeOS account, or use our Google account. We chose to use our Google account, locked down with two-factor authentication. Just a few minutes later, all the apps and settings from our Chromebook were ported to FydeOS. Even our *Chrome* bookmarks and history were synced to the FydeOS Pi 5. *Chromium* YouTube video playback was impressive at 1080p. We played a minute of *Big Buck Bunny* at 720p 60fps and in that time we saw only 95 of 3,311 frames dropped – much better than Raspberry Pi OS. However, the video stream did stutter and loop segments of audio when closing the browser.



Android apps appear as native applications in the **FydeOS** desktop, just like they do on **C**hrome**OS**.

When in Chrome...

FydeOS feels exactly like a ChromeOS device. We had the same user interface, without the Google branding. The app (start) menu is in the same place, we have pinned apps in the centre of the system tray, and quick access to settings, Wi-Fi and Bluetooth via the bottom-right menu. It may look like ChromeOS, but does it behave like a ChromeOS device? Let's find out.

Apps can be installed using the *Store* app, and we can even install and run Android applications, including Google apps via open GApps. We installed the Twitter (now X) app and sure enough we had a stream of tweets appear while we wrote this very review on the device. Android apps can be pinned to the FydeOS desktop tray for quick launch.

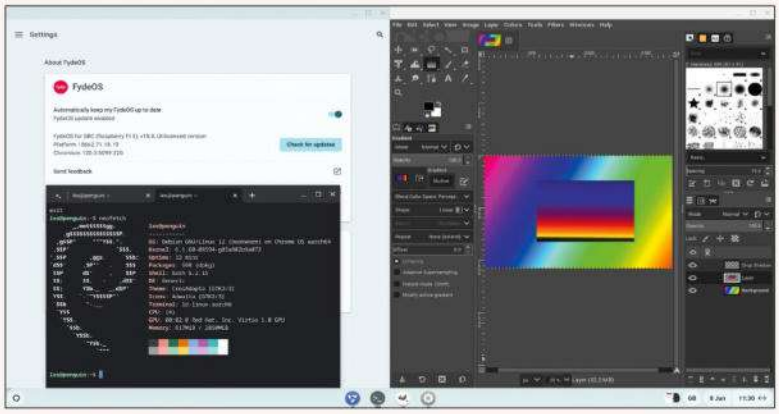
It wouldn't be *Linux Format* without us installing Linux on a device, and FydeOS provides the same Linux virtual machine service as ChromeOS. After following the configuration process, which downloads around 500MB of data and uses 10GB of storage, we had a Debian container ready for use. Here is the best bit: just like ChromeOS, we can install apps in the Linux VM with *APT,* then pin the apps to the FydeOS desktop. So, we have access to FydeOS/Chrome, Android and Linux apps!

This isn't an OS for GPIO or electronics projects on the Pi. For that you are best sticking to Raspberry Pi OS or Ubuntu. FydeOS is for light duties only.

It's not all plain sailing, though. Even with the power of the Pi 5's BCM2712 SoC Arm Cortex-A76 64-bit CPU at 2.4GHz, the system did hang. There were also crashes when we switched the KVM between our Linux desktop machine and the Pi 5 running FydeOS. And when we set the screen resolution to 720p to get a clear screenshot, it crashed and rebooted. These crashes aren't great, but can be fixed via software updates.

It's great for basic duties. If you love ChromeOS, FydeOS and a Pi 5 can provide the full experience. LXF

FydeOS provides the same Linux **VM** setup as **C**hrome**OS**, enabling us to have Linux, **C**hrome and **A**ndroid apps running on one desktop.



VERDICT

DEVELOPER: Fyde Innovations
WEB: https://fydeos.io
LICENCE: Mixed

FEATURES	8/10	EASE OF USE	8/10
PERFORMANCE	8/10	DOCUMENTATION	7/10

A fun alternative OS for your Pi 5, FydeOS needs a little bug squashing and performance tuning but it's a great experience.

» Rating 8/10

Anycubic Kobra 3

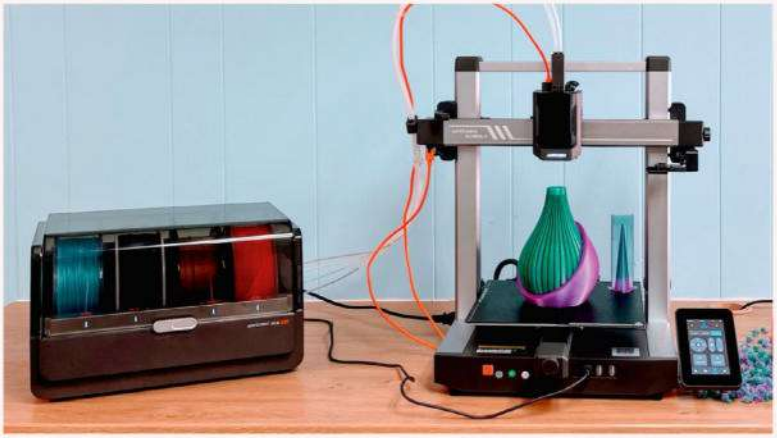
Any excuse to use a Star Wars reference, diminutive green-skinned **Denise Bertacchi** says begun the colour clone wars have!

SPECS

Build: 250x 250x260mm
Type: PLA/ PETG/TPU (up to 300°C)
Extruder: Direct drive
Nozzle: 0.4mm high flow
Plate: PEI spring steel, heated
Bed: Inductive auto levelling with Smart Z
Sensors: Runout, clogging
Comms: USB, Wi-Fi, cloud
Control: 4.3-inch touchscreen
Size: 452.9x 504.7x483mm, 9.2kg

Anycubic has caught up with Bambu Lab's multicolour system with a speedy four-colour bedslinger of its own, the Kobra 3. Although many 3D printer companies have been teasing four-colour AMS-like systems, Anycubic is the first to get its machine in customers' eager hands.

Mechanically, the Kobra 3 is on a par with the Bambu Lab A1. Its print volume is a few millimetres smaller, while its speed is a hair faster. It also has auto bed levelling and input shaping, with an onboard accelerometer. Like the A1, the ACE filament holder has a dedicated Bowden tube for each spool of filament, with a switching hub directly on the printer. The ACE has the added perk of being a heated filament drier, with automatic settings for many types of filament – even TPU, although TPU can't run through the ACE without tangling.



Go big or go home! Multicolour prints demand a bigger bench.

The biggest problem with the Kobra 3 is the slicer, which launched without any means to tune the waste purge flung from the printer while swapping colours. Like Bambu Lab machines, the Kobra 3 cuts filament before retracting it, which creates sizable squiggles of wasted filament to be expelled before a new colour is loaded. The Anycubic slicer lacks any controls for this or creating a purge object.

We set up the Kobra 3 in about 20 minutes. It ships mostly assembled, and you only need to mount the hotend and screen. Then you need to plug in the cables and connect the tubes.

Making the bed

The Kobra 3 has updated its automatic bed levelling system so that it first cleans the nozzle on a wiper brush attached to the build plate, then physically taps the bed and finds the correct nozzle height. This is automatically done during setup, along with calibration for vibration compensation (aka input shaping) and PID tuning.

On our test unit, the nozzle height was not accurate and had to be adjusted in the slicer. There is no way to manually adjust the nozzle height on the printer itself, which is extremely awkward. We asked Anycubic if this function would be added and was told it would be included in a future firmware update.

Anycubic brand filament has RFID tags to identify the filament type and colour automatically. When using third-party filament, you can manually enter the filament information from the printer's screen or the slicer suite.

The front plate is marred by a rubber plug to cover the ACE cable and USB ports. This is odd as it's a main selling point, which means this cover is left dangling open all the time. The printer has a quick-swap nozzle similar to what we see in the Bambu A1, which can be removed without tools. This is a one-piece nozzle inspired by E3D's Revo. It is specially made by Anycubic. The machine also has sensors in the tool head-mounted buffer to detect nozzle clogs. When it runs into trouble, it pauses the print and alerts you via the Anycubic app.

The Anycubic Kobra 3 comes with a copy of *Anycubic Slicer,* which is based on *Prusa Slicer.* The menus are arranged differently, so if you're familiar with *Prusa Slicer,* you have to poke around to find things. The slicer is missing a few functions and another issue is that *Anycubic Slicer* does not understand Bambu's 3MF files, which is how precoloured files are saved by designers. This means that if you want to print in colour, you need to paint the colours in *Anycubic Slicer* yourself.

Running a Speed Benchy with default rules, it didn't crack the top 10, which is odd considering its stats. This leads us to believe something hidden in the slicer is holding it back from its true speed. Our boat was a little rough, but its shape was well defined, with no signs of ringing. It was printed in ordinary grey Inland PLA, so none of the defects were hidden. LXF

VERDICT

DEVELOPER: Anycubic
WEB: https://uk.anycubic.com
PRICE: £449

FEATURES	7/10	EASE OF USE	7/10
PERFORMANCE	7/10	VALUE	7/10

A very good bedslinger with multicolour capabilities, hampered by a half-baked proprietary slicer.

» Rating 7/10

Raspberry Pi 5 AI Kit

Trying to find some intelligence on Earth, **Les Pounder** fires up the official AI kit for the Raspberry Pi.

SPECS

Support: Pi 5
Form: Pi M.2 HAT+
NPU: Hailo-8L Entry-Level AI Accelerator, 2242 package
TOPS: 13
Support: TensorFlow, TensorFlow Lite, Keras, PyTorch and ONNX
Host: x86, Arm

Based around the Hailo-8L entry-level NPU (neural processing unit) AI accelerator and the Raspberry Pi M.2 HAT+, this kit provides all the hardware that a Raspberry Pi 5 would need to beef up its AI powers.

We got our hands on an early unit, upon which this review is based. But before we begin, let's caveat the review. At the time of writing, the software was behind the rather excellent hardware, and while we were able to run the preview demos, we were unable to write any code of our own. As such, we'll be revisiting this kit once the software is updated.

A classic Hello World AI task of detecting the banana and human.

Installation and setup

Using the Raspberry Pi M.2 HAT+ board is a smart move. It already exists, and it has the connection and bandwidth for the Hailo-8L board. It also means that it comes with plastic spacers and a GPIO passthrough, which were a bone of contention for us in our previous review. Installation of the Hailo-8L is as easy as installing an M.2 NVMe SSD, but it does introduce one issue: unless you have a board with dual M.2 connectors, you can only run the AI or an NVMe SSD. Not both. Fear not, we shall be testing a dual M.2 board later, and (spoiler) it works if you put the effort in.

Our review kit came with the unit, Raspberry Pi Camera Module 3 and a microSD card that had the latest Raspberry Pi OS release written to it. It also contained a directory of models and applications based around the *rpicam* application suite. For those who don't know, the *rpicam* suite replaced the *raspicam* and *raspitill* terminal commands that are used to interact with the camera.

Our assumption is that a Raspberry Pi OS update will enable these features. At the time of writing, there are no *PiCamera v2*-based means to use the Hailo-8L and we confirmed this with Raspberry Pi.

Could we run the models without the Hailo-8L card? We asked Raspberry Pi CEO Eben Upton and he confirmed that there is "currently no way to run these models without the board". Upton did reveal that "integrated support for running unaccelerated TensorFlow models as a post-process is in the *rpicam* roadmap", so in the future we should be able to run our own models.

Using the *rpicam* suite of applications, we were able to try out a series of models that centred around the Raspberry Pi Camera Module 3. The *rpicam-hello* application is used as a demo. Passing it the correct series of switches (arguments) enabled us to try out the models without writing any data to the drive.

We tested four object-detection demos that identify an object and highlight using a bounding box. A label appears in the bounding box with what the AI thinks it can see, and a percentage to indicate its confidence. MobileNetSSD, Yolov5, Yolov8 and YoloX were the models and all apart from MobileNetSSD ran with an input resolution of 640x640; MobileNetSSD ran at 300x300. MobileNetSSD was the overall fastest, given that it was running at half the resolution of the others. While we can't quantify the exact speeds, we can give a 'feeling' for their performance. MobileNetSSD did feel the fastest, and the reduction in input resolution was not translated into the preview window. In fact, all of the object detection models ran remarkably well, but without a frame of reference, CPU performance, we can't provide any accurate data.

We tested out the image segmentation demo that uses Yolov5 with an input resolution of 640x640 at 20fps. This model detects objects and highlights them using a coloured mask. It accurately detected a person in the shot, and even our 3D-printed banana.

» DUAL NVME

You should spot that the Raspberry Pi M.2 HAT+ board has only one M.2 connector. If you've made the jump to fast NVMe storage, you don't really want to go back to microSD card storage. Even the best microSD cards for the Raspberry Pi are not as fast as NVMe, and while a fast drive won't help your AI projects in general, it helps.

We tested Pimoroni's NVMe Base Duo with the Hailo-8L and a Phison-based 2230 NVMe drive left over from a recent Steam Deck upgrade. The NVMe drive was placed in slot A, Hailo in slot B. After updating the Pi 5 firmware to 17th May 2024 via microSD, we booted from NVMe. To our delight, it just worked. Raspberry Pi OS booted in seconds. The latest firmware gives boards with dual M.2 connectors the ability to boot from the first drive, leaving the second M.2 slot free for more storage or, in our case, using the Hailo-8L. PCIe Gen 3 speeds are also enabled by default – a nice bonus.

The final demo was pose estimation, and this estimates up to 17 points of articulation, including face tracking it seems. We fired up the demo and used it to track our pose. Hands, arms, shoulders and head were tracked from our sitting position. We could see a wireframe to represent our posture and from that we could control it like a puppet.

The demos worked, but what now? As it stands, at the time of writing there isn't a clear-cut way to use the data. Ordinarily we would use the data with an application to react. For example, a Python application that triggers actions based on identifying objects, or triggering motors using poses.

Because there is currently no *PiCamera* support, we can only really use the *rpicam* suite of applications. There is a **-v 2** argument that we can pass to these commands, which then writes the output to the standard output. With a little Linux-fu, we can then store this information in a file, to be parsed into another application. We could also run the command using the os.system or subprocess Python modules and interact directly. The choice lies with the user.

Before we attempted one of the demos, we first checked that Raspberry Pi OS could see the board. Using *dmesg* and piping the output to *grep,* we sliced out all references to 'hailo' and we could see the device was listed. Buoyed by that news, we fired up one of the demos and saw ourselves appear on the screen. Thankfully, we were classed as a person by AI, and our 3D-printed banana for scale (printed on the Elegoo Neptune 3 Pro) was correctly identified.

Obviously, there are people who want to integrate AI into their projects. We can see this being used in citizen science projects, classrooms and for robotics projects. At the recent Pi Wars event, AI was on the cusp of being usable for many robots. The Dutch Rescue Team used OpenCV to help its robot identify and follow a line across a treacherous course. For science projects, it could be used with a model that identifies birds, insects or animals that visit your garden. For those who want to embed AI in their projects and products using the Raspberry Pi, this combination could be an interesting avenue to explore.

The AI accelerator mounted on the M.2 connector and ready to take over the world.

Gesture detection is one of the out-of-box demos you can try.

We enjoyed our time with the Raspberry Pi AI kit, but it wasn't plain sailing. At the time of writing, the software is not to the same level as the hardware. The hardware 'just works' largely down to running Linux kernel 6.6.31. Raspberry Pi OS understands that the board is there, and it has the firmware to use it. Once the software, in the form of *PiCamera v2* and *rpicam,* catches up, we'll revisit this board and put it through its paces. For now, this is a board for those who want to dabble in AI or for those who know their stuff and want to delve deep. It's not a board for everyone, but it could be soon. LXF

If you want to keep your boot NVMe drive, you need an adaptor.

VERDICT

DEVELOPER: Raspberry Pi
WEB: www.raspberrypi.com/products/ai-kit
PRICE: £65.70

FEATURES	6/10	EASE OF USE	8/10
PERFORMANCE	8/10	VALUE	7/10

A great piece of hardware, but until the software updates come through, it can't be fully utilised.

» Rating 7/10

PI PICO

How to make more sensor-ble choices

When not writing terrible puns, **Les Pounder** delves into the myriad of sensors that you can use with the Raspberry Pi Pico.

OUR EXPERT

Les Pounder is associate editor at Tom's Hardware and a freelance maker for hire. He blogs about his adventures and projects at **http://bigl.es**.

YOU NEED

- **Pico/Pico W**
- **HC-SR04P or HC-SR04+ ultrasonic sensor**
- **Obstacle sensor/PIR**
- **RCWL-0516 sensor**
- **Half-size breadboard**
- **4x M2M jumper wires**
- **3x F2M jumper wires**
- **Linux PC**
- **Code: https://github.com/lesp/LXF318-Sensors/archive/refs/heads/main.zip**

We're going back to basics and looking at a trio of cheap sensors that can be used in a plethora of projects. The goal is to learn how sensors work and how we can integrate them into our Raspberry Pi Pico and Pico W builds. We're using *Thonny* to write the project code.

Using a PIR sensor

The passive infrared sensor is the most basic sensor input. The BISS0001 sensor is used in many PIR sensors and has only two states. When idle, the output pin is at 0V, but when movement is detected, the pin pulls to 3.3V. This change of state is our trigger.

In a new file, start by importing the machine and utime modules. Machine is used to interact with the GPIO, while utime is used for precise timing.

```
import machine
import utime
```

Create an object to act as a link between our code and the PIR sensor connected to the GPIO at pin 16. Then set the pin as an input, and pull the GPIO pin down, so that it is at zero volts. This ensures that the pin can be read correctly.

```
PIR_PIN = 16
pir = machine.Pin(PIR_PIN, machine.Pin.IN, machine.Pin.PULL_DOWN)
```

Create an object called **counter**, which we'll use to count the number of sensor triggers. Normally we would start counting from zero, but because we are counting occurrences, we start from one.

```
counter = 1
```

Print a start message to the Python shell:

```
print("Starting PIR sensor monitoring...")
```

Using a **while True** loop, we run the contained code for ever, starting with a conditional test that checks the state of the PIR pin. If there is movement, the pin pulls high (on, True, 1) and this is our trigger.

```
while True:
  if pir.value() == 1:
```

The movement triggers a message to the Python shell, formatted using Python's f-strings so the counter value is inserted into the message. We then increment the counter by one, ready for the next trigger.

```
    print(f"Motion detected, {counter} times")
    counter += 1
```

While the PIR sensor detects movement, we need to wait, otherwise it triggers a false detection.

```
    while pir.value() == 1:
      utime.sleep(0.1)
```

Finally, outside the conditional test but inside the main loop, we tell the code to wait for 0.1 seconds before repeating the loop.

```
  utime.sleep(0.1)
```

Save the code to the Raspberry Pi Pico as **pir.py** and then click on the green run icon to start the code. Point the sensor away from yourself and wait for it to settle. Now wave your hand in front of it and watch the Python shell as it counts the number of triggers.

Alternative sensors

If you wish to use an infrared obstacle sensor, you need to modify the code. The resting state of the sensor is HIGH, so we need to set the **pir** object to pull the pin UP. When the sensor is triggered, the state changes from HIGH (1,True, UP) to LOW (0, False, DOWN)

```
pir = machine.Pin(PIR_PIN, machine.Pin.IN, machine.Pin.PULL_UP)
```

Also change these **pir.value()** lines to check for 0.

```
  if pir.value() == 0:
AND
    while pir.value() == 0:
```

Alternatively, we can use an RCWL-0516 Radar sensor to detect movement. These cheap sensors are microwave proximity switches and great fun. They use the same logic as the PIR sensor, so we can just swap them out. Why are they fun? They can be embedded inside a project, as the detector doesn't rely on light, rather it uses microwaves to detect movement. Save the code to the Pico as **microwave.py** and click Run.

Ultrasonic sensors

Looking like a mini Johnny Five from the *Short Circuit* movies, the HC-SR04 is a classic sensor. Commonly used in robotics builds because they are cheap and accurate (+/- 0.5cm in our tests), these sensors use a pulse of ultrasound to measure distances. There are many versions, but we recommend the HC-SR04P and the HC-SR04+ models as they work with 3V and 5V logic. The older HC-SR04 only works at 5V logic; great for Arduinos, but bad for the Raspberry Pi's 3V GPIO.

Using the HC-SR04P/+ with the Raspberry Pi Pico is simple. In a new file, start by importing the machine and utime modules as before.

```
import machine
import utime
```

Create two objects that represent the GPIO connections between the Pico and the HC-SR04. The trigger of the sensor (sends a pulse) is connected to GPIO3, the echo (receives the pulse) to GPIO2.

```
TRIG_PIN = 3
ECHO_PIN = 2
```

We need to set up the trigger and echo pins on the Pico so the trigger is an output and the echo an input.

```
trigger = machine.Pin(TRIG_PIN, machine.Pin.OUT)
echo = machine.Pin(ECHO_PIN, machine.Pin.IN)
```

Create a function called **measure_distance** and inside the function we need to set the trigger pin low (off) before sleeping for two microseconds. Then set the trigger to high (on) for 10 microseconds, before turning the trigger pin off. We've just written the code to send a pulse of ultrasound from the trigger pin.

```
def measure_distance():
    trigger.low()
    utime.sleep_us(2)
    trigger.high()
    utime.sleep_us(10)
    trigger.low()
```

Next we need to write the code that waits for the return of the pulse, the echo. Two **while** statements check the status of the echo pin. The first sets the **pulse_start** time and records the time in microsecond ticks. When an echo is received, the **pulse_end** object stores the current time in microsecond ticks. The ticks are a counter between the start and end of the pulse.

```
    while echo.value() == 0:
        pulse_start = utime.ticks_us()
    while echo.value() == 1:
        pulse_end = utime.ticks_us()
```

We've got the start and end times, so now let's do the maths. The **pulse_duration** is the difference between the end and the start of the pulse.

```
    pulse_duration = utime.ticks_diff(pulse_end, pulse_start)
```

We've got the duration, now we need to work out the distance. Distance is time multiplied by speed but we need to divide the answer by two. Why? Because the **pulse_duration** measures the time to the object and back. The speed of sound is 34,300cm/s. We then round the returned distance to two decimal places and return the value.

```
    distance = (pulse_duration * 0.0343) / 2
    distance = round(distance, 2)
    return distance
```

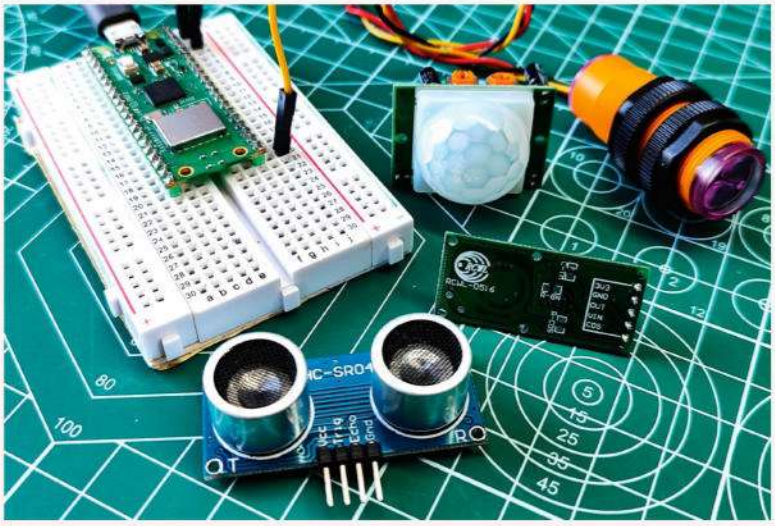
The function is complete, so let's use it. Inside a **try/except** block, we try to continually run the **measure_distance** function, saving the returned distance to an object called **distance**. Using Python's f-strings (a way to format data inserted into strings), we drop the distance measurement into a short message. We then pause the code for a second before the loop repeats.

```
try:
    while True:
        distance = measure_distance()
        print(f"Distance: {distance} cm")
        utime.sleep(1)
```

Should we need to exit the code, the **except** section is activated, printing a message to the Python shell.

```
except KeyboardInterrupt:
    print("Exiting")
```

Save the code to the Pi Pico as **ultrasonic.py** and click on the green run icon to run. The Python shell shows the approximate distance every second. LXF



All of these sensors are low cost and can be used with the Raspberry Pi Pico, Raspberry Pi and many other boards.

» SENSOR CHOICES

Choosing the right sensor largely comes down to what data you want to collect. We kept it simple for this tutorial. We want to know where things are. The PIR sensor detects movement in a broad 180° path. The obstacle sensor can detect objects in order to avoid them (typically around 5-10cm). The RCWL-0516 microwave doppler sensor is like the PIR sensor but it can be hidden inside a build. The most precise sensor in this tutorial is the HC-SR04 as it is designed to detect objects at a distance. With a little maths, we managed to determine the distance from an object with reasonable accuracy.

For a robotics project, which would be best? The first answer is the HC-SR04. This has been used to give a robot the ability to sense the world around it, and with some clever coding, we can navigate a maze. The next choice would be the obstacle sensor, which provides a short-range ability to detect an object. You could use this to navigate a maze, but we'd use it in conjunction with an HC-SR04 as a means to keep the robot from getting stuck.

A line-following sensor, like Cytron's Maker Line, has five line-following sensors that look for dark or light lines and keep the robot on the path by ensuring the central sensor can see the line. Deviation is detected by the outside sensors and coding moves the robot back to the path.

RASPAP

Credit: https://raspap.com

Smart Pi Wi-Fi

Press-ganging a Raspberry Pi into access point duty helps **Tam Hanna** use a wide variety of functions, from virtual relocation to ad blocking.

OUR EXPERT

Tam Hanna has been working with computers, software and electronics for more than 20 years.

While everything we discus here can be done on the client, putting it into an access point is beneficial. Deploying stuff can be done without administrator rights on the target; the device's owner connects to the wireless network established by the Raspberry Pi and automatically benefits.

The features found in *RaspAP* are mostly part of the underlying operating system. *RaspAP's* added value is a convenient way to a guaranteed working configuration; furthermore, the access point exposes a web interface for administration.

RaspAP works best with Raspberry Pi OS, but the list at **https://raspap.com/#distros** includes a few other options. In the interests of commonality, the following steps take place on a Raspberry Pi.

We use the image **2024-03-15-raspios-bookworm-arm64.img.xz**. Evidence points to the desktop-less version of Raspberry Pi OS Lite working equally well, but we are using the full version.

Use a high-quality power supply – creating a Wi-Fi network while powering the power-hungry Ethernet port is a sure-fire way to experience power problems.

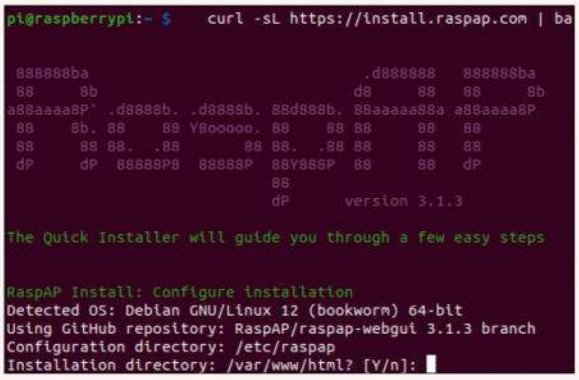
Getting started with Raspberry Pi requires a completely updated operating system. Run the following two commands and perform a reboot:

```
$ sudo apt-get update
$ sudo apt-get full-upgrade
```

Due to reasons of localisation, the Wi-Fi transmitter on the Pi is unlocked by geographic information. This issue can be remedied by launching the configuration utility via `sudo raspi-config`, and selecting Localisation Options and WLAN Country. Then select the Wi-Fi country. Afterwards, close the configuration tool and reboot the Raspberry Pi to finalise.

Actual installation is accomplished via a Bash script, which can be downloaded from the *RaspAP* repository.

Decent speed can be expected on a Raspberry Pi 4.



This configuration system leads to a working access point.

Enter the following command in the installation wizard, which works in a semi-graphic way, similar to the one shown in the screenshot *(above)*:

```
$ curl -sL https://install.raspap.com | bash
```

The first step asks about paths for various software components. The defaults can be accepted by repeatedly pushing hitting Y (as is tradition) and Return. During the script's execution, an additional `apt-get update` is performed – this is normal, as the script has to add package sources to the system. Locale-related warnings that pop up during the installation of PHP constitute a non-issue.

After the installation of PHP has been completed, the system asks about some configuration settings. In particular, options such as Enable Http Only For Session Cookies (Recommended)? can be accepted at the default value.

People intending to activate ad-blocking are advised to push Y at the Install Ad Blocking And Enable List Management? prompt. While the settings can later be added, making sure the configuration is created by the installation script reduces grief later on during the product life cycle.

In the interest of simplicity, further options such as OpenVPN are not enabled for now. Instead, permit the script to perform a reboot of the access point. The Raspberry Pi's graphical user interface is disabled – from now on, *RaspAP* only boots up to the text-based log-in interface.

Access to the point

When the Raspberry Pi has rebooted, your Wi-Fi is visible. By default, the router has the settings shown in the following table:

```
IP address: 10.3.141.1
Username: admin
Password: secret
DHCP range: 10.3.141.50 — 10.3.141.254
SSID: raspi-webgui
Password: ChangeMe
```

As a first test, *Ookla Speedtest* is a good measure. These experiments were performed against a gigabit internet connection provided by Magyar Telekom – the results *(see screenshot, facing page, bottom)* show the speed is good enough for most lightweight scenarios.

The most interesting aspect of the configuration is the IP address of the event access point. Enter it into a machine connected to the Raspberry Pi Wi-Fi to bring up the configuration interface.

The URL **http://10.3.141.1** can be opened in a browser. Using the web interface is recommended – it permits you to configure various security-related options. Changing both the name and the password of the Wi-Fi is recommended to keep miscreants out – the defaults are known to even the newest of noobs, and script kiddies have attacked *RaspAP* in the past.

Interestingly, the ad-blocking feature is not very successful when dealing with YouTube – in our tests, a reduction of about 50% of ads was discernible. On most other sites, more radical results were possible.

Adding VPN support

When working from home, being able to fake your location can be helpful – nobody needs to know whether an email was composed in Germany or on a Caribbean island. While using a VPN is not difficult, its installation requires administrator rights.

Advanced-level system-monitoring solutions (especially on Windows) can detect the presence of a VPN and alert the employer. Using a Raspberry Pi works around the problem – as the traffic gets tunnelled at the access point level, detecting the use of a VPN is more difficult (albeit not impossible).

A second pass is required because we did not select the option when running the quick install. In practice, selecting all options can be a better approach – disabling unwanted ones in the web interface is straightforward. Rerunning the command above forces you to rework all of the existing settings.

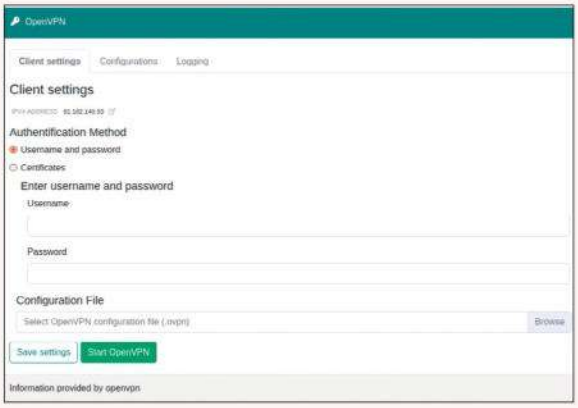
In theory, missing components can be deployed by following the instructions at **https://docs.raspap.com/manual**. More comfortably, the script can be rerun using parameters outlined at **https://docs.raspap.com/quick**. In the following steps, we install VPN support:

```
$ curl -sL https://install.raspap.com | bash -s -- --yes --provider 0
```

Provider-specific configurations tend to work poorly. Thus, parameter **provider 0** is passed on to enable VPN support without selecting a provider.

Given that the installation script is directly passed into Bash, the parameters are set by the sequence `bash -s -- --yes`. Passing in **yes** is helpful and instructs the installation script to assume valid defaults.

After running the installer, a reboot is recommended – the script does not push for a reboot. Simply attempting to continue using the access point's web interface leads to a situation where the newly downloaded features are not available. When rebooted, the interface provides the new set of options *(see screenshot, above)*.



This option allows configuration of the VPN capabilities.

The section for uploading OVPN files is of particular interest. OVPN files are a standardised way for VPN providers to express settings needed for protection.

The exact method for generating these files is dependent on the provider of choice – in the case of NordVPN, for example, you need to log into the back-end and then use the option Manual Setup > Get Service Credentials Or An Access Token.

As well as credentials, the OVPN file identifying the server to be used has to be downloaded from **https://nordvpn.com/ovpn/**. After that, the access point can be parameterised using the settings window.

To activate the settings, click on Save Settings. The interface then refreshes, after which a click on the green Start OpenVPN button is recommended.

After clicking the button, the web interface appears and a VPN checker confirms that the VPN is active.

While a homemade access point based on a Raspberry Pi might not be quite as fast as a purchased one, using *RaspAP* provides a great variety of configuration options. This short tutorial acts as a first introduction, so feel free to explore deeper! LXF

QUICK TIP

If your access point is for the benefit of the technically challenged, a robust 3D-printed housing increases longevity. Visit www.thingiverse.com or https://models.makewithtech.com to find 3D-printable templates.

» RASPAP INSIDERS

Even though *RaspAP* is under the GPL licence, additional features are available to paying customers via the Insiders program. It uses the GitHub sponsoring feature – visit **https://github.com/sponsors/RaspAP** and log into your GitHub account to set up a sponsorship at a minimum of $10 per month.

Once you've set up your sponsorship, your account is granted access to a dedicated repository. It houses the advanced components of *RaspAP*, which are not available in the standard version. By using the special installer provided, you can unlock a variety of features that significantly enhance your experience.

One interesting design decision is that features become available as funding goals are met. A breakdown of the goals and the amount of money still to be raised can be found at **https://docs.raspap.com/insiders/#500-1st-insiders-edition**. However, giving money to an important project is always a good idea, especially if you derive long-term value from using this system.

Part Three! Don't miss next issue, subscribe on page 16!

Inside Linux The display

CREDIT: Magictorch

Matt Holder delves into the history of our display servers and introduces consoles.

In this month's article, we are continuing our investigation of the internals of Linux distributions. In the last two articles of the series, we looked at the Linux kernel and discovered some of its capabilities, such as interacting with hardware, providing firewall functionality and a virtualisation platform, and much, much more. While we recommend reading the first two instalments, the content in this article can be considered as standalone and no prior knowledge is required.

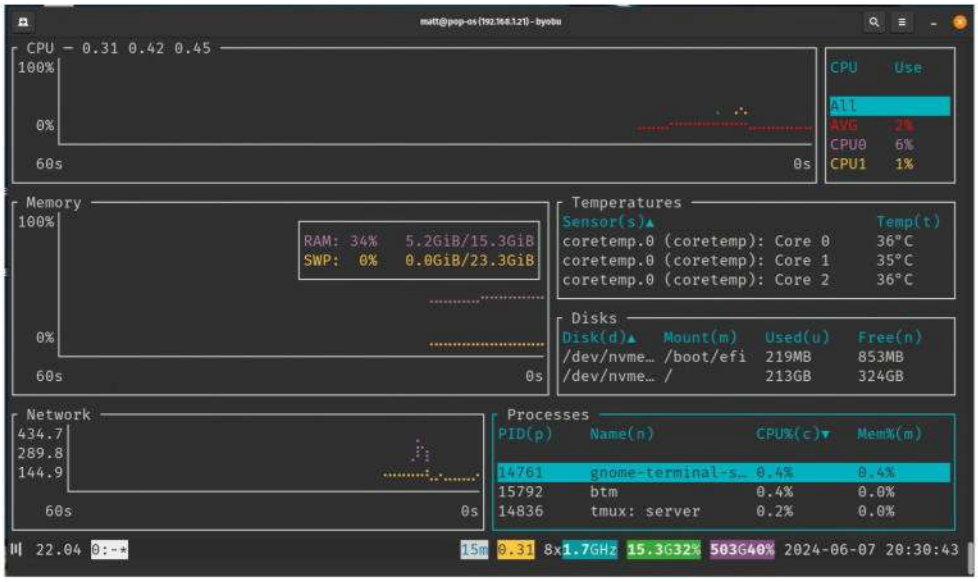
This month, we are discovering the various ways in which we can interact with our distributions, from the humble console to the venerable X server, to the Wayland protocol, which has multiple implementations, including the Weston reference compositor.

The time of terminals

In many of our articles, you need to open a terminal application and run commands, which perform a task and then display output on the screen. This was one of the earliest ways to interact with the system and it

takes time to learn the commands. If we go back in time, to the dark old days before we had internet access, learning about the commands we needed to use was not as simple as it is now – you probably had to visit your local library or book shop to borrow or purchase a book about the Linux operating system.

When using a terminal from a graphical user interface, this can be referred to as a virtual console. What we will discuss in the next paragraph is interaction with a console. This is not a historic skill to learn – many Linux servers are headless, meaning that access is via a protocol, such as SSH, and commands need to be run.



The Byobu terminal multiplexer running btm.

Before getting into display servers, let's pretend we're in the late '90s or early 2000s. Before carrying out the following steps, ensure everything is saved and then press Ctrl+Alt+F3. This drops you to a console, which allows you to log in with your usual credentials and enter some commands. Long-running commands can be suffixed with `&`, which runs the command in the background and allows another task to be started as well. Experiment with running commands in the background and you'll notice that you won't see any output while the command runs. If you want to run multiple long-running tasks and see the output, you can press Alt+F2, Alt+F3 and so on, and this gives extra login prompts that you can use to log in. Switch between them with the Alt and F keys. Below is a list of commands that you can experiment with and enjoy the good old days.

Search the current directory, and sub directories for files that start with test:

```
find . -name "test*"
```

View the man (manual) page for the find command:

```
man find
```

View the space of any mounted disks or partitions:

```
df -h
```

View the size of files and folders in the directory:

```
du -ch .
```

On a Debian / Ubuntu based system, install the ncdu command:

```
sudo apt install ncdu
```

Launch ncdu to view space usage in all directories:

```
sudo ncdu /
```

Change directory to your home directory:

```
cd ~
```

Change directory to the root directory:

```
cd /
```

In the above examples, when running the `sudo` command, we are running the command as the root or super user. Sudo stands for Super User Do. There are many other commands that you can investigate, including `ip`, which is used to interact with the network interface. When you have finished interacting with the consoles, press Ctrl+Alt+F2 and you will either be back where you left off or sitting at the graphical login screen, ready to log in again. These instructions were written for Pop!_OS, which is running the display under Wayland. Instructions may change slightly based on your distro and whether you are running Wayland or X.

Too many terminals

When using a terminal, it is possible to run multiple commands at the same time and even disconnect from a session, which can be reconnected to at a later date. Multiple tools exist, such as *GNU Screen, TMux* and *Byobu*. The latter was developed by Canonical as an extension of the aforementioned tools. *Byobu* allows elements to be written to the terminal, such as CPU or memory usage, disk space and the date and time. Using keypresses, you can create new 'screens' and move between them. On an Ubuntu-based system, to install and run *Byobu*, open a terminal and run the following commands:

```
$ sudo apt install byobu
$ byobu
```

Once loaded, you will see that your terminal has been transformed and shows lots of useful information. Run a command and it displays to the screen as usual. By pressing F1, *Byobu* creates a new terminal prompt for you to use. Pressing F2 or F3 enables you to move forwards or backwards between terminal prompts. Pressing the F6 key disconnects you from the session, which you can connect to again by running *Byobu* again. Using the F8 key enables you to rename the window, and Shift+F2 or Ctrl+F2 splits the terminal into multiple sections, so that multiple commands can be run and the output seen on one view.

As the capabilities of computers improved, it became possible to display graphical items on the screen rather than just text. Compared to today's specs, imagine being one of the original programmers, who needed to optimise the code so that every CPU cycle and byte of memory was used efficiently.

X marks the server

In 1984, the X Window System started development. This was part of Project Athena at MIT. Project Athena was run by MIT, DEC and IBM to provide MIT with a campus-wide computing environment. In 1987, the protocol used by X reached version 11, and from then on, X has been known as X11. X provides an

QUICK TIP

While not known as the most reliable site in the world, you can learn more about X11 here: https://en.wikipedia.org/wiki/X_Window_System

```
Section "ServerLayout"
        Identifier     "XServer Configured"
        Screen      0  "Screen0" 0 0
        InputDevice    "Keyboard0"  "CoreKeyboard"
        InputDevice    "USB Mouse"  "CorePointer"
EndSection

Section "Monitor"
        Identifier   "Monitor0"
        Option       "DPMS"       "true"
        HorizSync    31.0 - 61.0
        VertRefresh  50.0 - 90.0
EndSection

Section "Device"
        Identifier  "Card0"
        Driver      "vesa"
        VendorName  "All"
        BoardName   "All"
EndSection

Section "Screen"
        Identifier "Screen0"
        Device     "Card0"
        Monitor    "Monitor0"
        DefaultColorDepth 24
        SubSection "Display"
                Depth     24
                Modes "1024x768" "800x600" "640x480"  "1600x1200" "1280x1024" "1280x960"
        EndSubSection
        SubSection "Display"
                Depth     32
                Modes "1024x768" "800x600" "640x480"  "1600x1200" "1280x1024" "1280x960"
        EndSubSection
EndSection
```

An example of an X server configuration file can be seen here.

environment in which graphical applications can be run. These apps can be standalone, such as *XClock* or *XEyes,* login managers, where you log in to your system, through to desktop environments, such as Gnome, Xfce, KDE and Budgie. The version of X server that we use originates from XF86, which began in 1991 as a port of the version developed as part of Project Athena. In early 2004, the X.org foundation was formed and over the next year or two the majority of development moved over from XF86 to X.org. The split between the two organisations solved many problems and meant development could carry on as required.

X MARKS THE SPOT

"Once the install has completed and you have rebooted your device, X starts and devices are queried to determine their capabilities."

X operates with a server and client model, which makes a lot of sense, given that Project Athena was a distributed computing environment, which would have had centralised computing and terminals to access the more powerful central system. This means that it is possible to run X applications over the network, either via a secure connection system, such as SSH, or other network methods. It is incredible to think that a piece of software that started development in 1984 and has been at protocol version 11 since 1987 is still able to fulfil so many of today's requirements.

By changing the configuration of your X server, you can allow remote connections from another server; while this is outside of the scope of the article, plenty of instructions exist online. To open an application from another device, which is running X and the SSH server, enter the following commands in a terminal.

```
$ ssh -X Username@IP_ADDRESS_OR_HOSTNAME
APPLICATION_NAME
```

Part of the reason that this is outside the scope of the article is that a lot of modern Linux distributions use Wayland instead of X, which means that your server will not be running.

As previously mentioned, X server has a rich history and its development has come a long way over the years. Installing a distro is generally as simple as booting from a USB device and using the installer. Once the install has completed and you have rebooted your device, X starts and devices are queried to determine their capabilities. For example, the Extended Display Identification Data (EDID) standard is used for displays to advertise their specifications. This has not always been the case, however, and configuration of the X server, with a textual configuration file, was originally needed. An example configuration file can be seen in the screenshot *(above-left),* which will be described shortly.

The first section in the example configuration file refers to a ServerLayout, which is used to group together a screen as well as any input devices. Next in the example is a section that defines a Monitor. This refers to the physical device and defines critical values, such as the Horizontal Sync and Vertical Refresh values. It is imperative that this is seen as an example and not used with your own hardware, because it is possible to damage certain types of monitor by setting the wrong values. The third section refers to a Device and this represents a graphics card, along with the driver that should be used. Finally in the example configuration file, we see a section for a Screen. This is used to link together the physical monitor, graphics card that should be used and the possible resolutions that can be selected. Using the configuration file, it was also possible to define extended or mirrored displays for multiple monitors or to start a separate X server on each display.

QUICK TIP

You can learn more about Wayland and the Weston reference compositor here: https://wayland.freedesktop.org

Way more Wayland

Now we can smoothly segue to Wayland and find out what it is. Over the years, if you've ever been using your Linux distros and wondered why you might be seeing strange artefacts, such as video tearing or missing frames, X may have been the cause. While the software has given amazing service over its 30-year history, there are some situations where it isn't able to meet our more modern requirements.

For example, if you have multiple monitors, where one has a higher DPI rating than the other, X is going to struggle to allow you to configure each differently and

have a smooth experience dragging applications between monitors. Another example where Wayland has been designed to shine is with smooth video playback and addressing security concerns that stem from the client-server nature of X.

So, what is Wayland exactly? Well, it is a protocol that is used to specify how a display server and clients (graphical applications) communicate with each other. This is similar to the X protocol, which is used by X servers. When the Wayland specification is realised as code, the display server is known as the Wayland compositor. This is because the compositor performs the tasks of both a traditional X server and a compositor. Compositing describes the process of adding extra effects to what is displayed on our screens, such as minimise and maximise effects, and everybody's old favourite effect of switching between applications using a spinning cube.

Security is heavily built in to the design of the protocol to mitigate some of the issues in the fundamental design of the X server. Wayland by itself does not have the capability to operate over the network in the same way that X can, but compositors are free to add remote connection support, such as VNC or RDP.

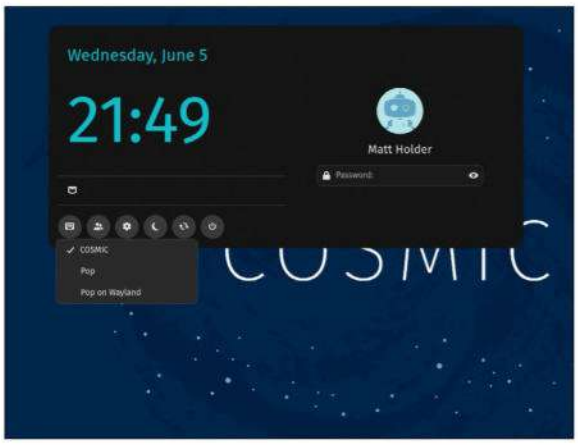
The Weston compositor is a reference implementation by the same team that develops the specification. There's a number of other compositors that can be used that all serve different purposes. Hyprland is a compositor used to provide a tiling window manager, which looks excellent. KDE and Gnome both implement their own compositors, called KWin and Mutter respectively.

Server selection

While there's a number of distributions that use Wayland as default, some of them also ship with an X server as well. It is useful to be able to swap between the two environments to see the differences, and if something is missing from one environment, you can easily move to the other one. When logging into your device, there is usually an option on the screen to select the session type. Using Pop!_OS, select or enter your username. Once this has been entered, click on the cog and then select the relevant option. Using the example in the screenshot *(above-right),* you can pick between Cosmic (which is currently in pre-alpha status and is the desktop environment designed and built largely by the team at System 76), Pop and Pop on wayland. If you have other desktop environments installed, such as KDE, XFCE or Gnome, these appear here as well.

Getting on with the X

There are significant differences between X and Wayland, and applications need to be developed accordingly. Applications that use toolkits such as *Qt* (KDE uses this toolkit) or *GTK* (Gnome uses this toolkit) are able to run on X or Wayland, as their toolkit has been designed to run in either environment. If you are using applications that either do not support Wayland directly or use toolkits that don't support both systems, thankfully all is not lost. XWayland is a project that takes an X server and runs it as a Wayland client. A good example of this is the venerable *XEyes,* which is written for usage on an X server and, when running, the pupils of the eyes follow our mouse cursor. Interestingly enough, due to the segregated nature of applications running under a Wayland compositor, the pupils only follow your mouse cursor when the application is in focus. When using the application under an X server, the pupils follow the cursor even if the application isn't in focus.



Multiple desktop environments can be selected from the logon screen.

QUICK TIP

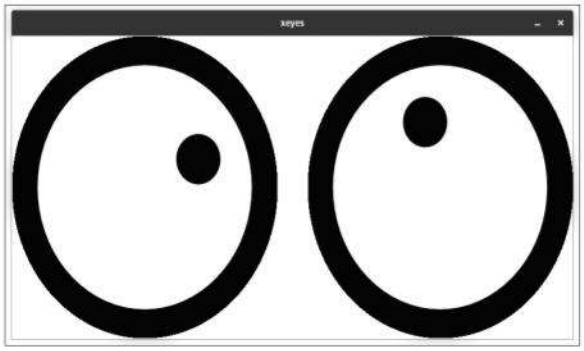
Keybindings for Byobu can be found here https://help.ubuntu.com/community/Byobu#Key_Bindings

Now that Wayland has been so brilliantly integrated into our distributions, it can be difficult to know which display server you are using. By opening a terminal and running the following command, you will be able to find out:

```
$ echo $XDG_SESSION_TYPE
```

There are lots of applications available that can be used to take screenshots of our desktops or applications we are using. Most desktop environments provide their own tool and these can be used with a Wayland compositor or an X server. The KDE project has developed a tool called *Krdp,* which provides the necessary glue to allow RDP access to the KDE Plasma environment. This is excellent, as remote access to a Wayland session wasn't possible at one time. Similar to the earlier example of using the `ssh -X ...` syntax to open X applications from a remote session, the *Waypipe* program can be used for applications running under Wayland (visit **www.mankier.com/1/waypipe#** to find out more).

We hope that you have enjoyed this whistle-stop tour of display servers and please join us again next month, when we will be covering the PipeWire project, which has been designed to greatly improve the handling of audio and video under Linux. In fact, the PipeWire project is used to allow the excellent *OBS Studio* tool to capture Wayland applications for recording or streaming. LXF



XEyes is an X application that can be used to track the location of your cursor.

JOPLIN

Credit: https://joplinapp.org

Sync terminal notes

Shashank Sharma is troubled by random thoughts. He now scribbles them down using Joplin. But what about that Portugal penalty shootout?

OUR EXPERT

Shashank Sharma is a trial lawyer in New Delhi and an avid Arch user. He's been writing about open source software for 20 years, and lawyering for over 10.

The beauty of note-taking apps is that they enable you to quickly put your thoughts down before they become lost in the sea of other thoughts. Linux users are spoilt for choice when it comes to note-taking applications, while some people prefer the comfort of a good old **notes.txt** file, created and continually populated with their favourite text editor. A good note-taking utility, however, can help you tag your notes and even search for particular notes. If you work across various platforms and devices, though, and want to keep your random thoughts in sync, you should try *Joplin*.

Joplin stands out from most of its peers because it supports various devices and platforms. You can install it on Linux, Mac and Windows, and there are apps for Android and Apple devices as well. The project supports various third-party cloud platforms, such as OneDrive, Dropbox, Nextcloud and so on, to keep your notes in sync across devices. Best of all, the CLI variant is just as featureful as the graphical offering.

Notes from nodes

Joplin is a JavaScript utility released under the AGPL v3 licence. You have to ensure your system has Node 12+ installed already. At any given time, there are various versions of Nodejs on offer. These are the current release, the LTS release and the stable release. There are various methods to install Nodejs on your Linux distro, but we can only discuss one due to space constraints in the magazine.

First you have to install the *Node Version Manager (nvm).* Run the following command:

```
$ curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.7/install.sh | bash
```

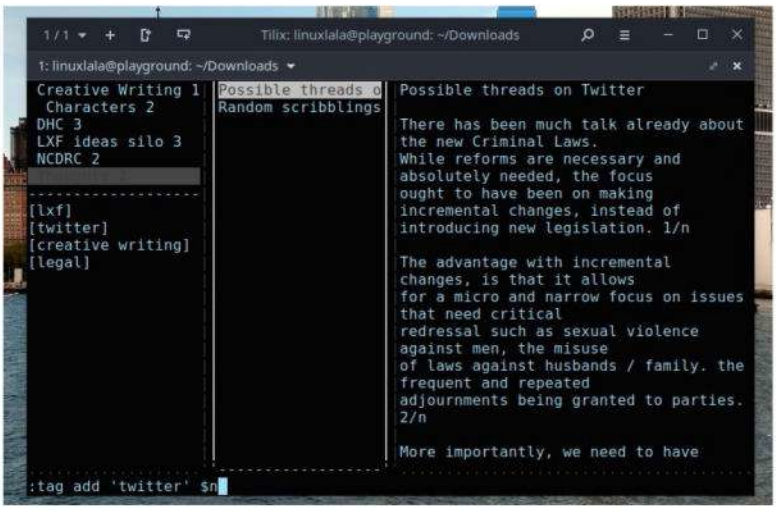
Once done, close and relaunch your terminal emulator. You can now run `nvm ls-remote` for a list of available Nodejs packages. Now run the `nvm install 20` command to install the 20.15.0 LTS version.

With the compatible version of Nodejs installed, we now need to configure *npm,* the Nodejs package manager, so that we don't need to use *sudo* when installing packages using *npm.*

The first step is to create a directory for packages. This could be **~/.npm-global** or **~/Downloads/.npm-global** or any directory of your choice. Next, run the `$ npm config set prefix '~/Downloads/.npm-global'` command to set the new directory path. You also need to update your **~/.bashrc** or **~/.profile** file, and add the `export PATH=~/Downloads/.npm-global/bin:$PATH` line. Finally, run the `source ~/.bashrc` command.

You can now install *Joplin* with the `npm install -g joplin` command.

When you add tags to your notes, the tags are also listed in the left-most pane, under all the notebooks.



Note it down

Joplin uses Markdown to help you quickly format text. You can refer to the Markdown Guide on *Joplin's* website to learn how to format your notes (**https://joplinapp.org/help/apps/markdown/**).

The utility organises notes into various notebooks. You can create as many notes as you like within each notebook, and can have multiple notebooks. When using the graphical desktop application, you can also create sub-notebooks, but this feature isn't available as yet on the mobile app or the CLI utility.

When you launch the utility with the `joplin` command, you're presented with a three-pane interface. The panes aren't named, but you'll get the hang of it quickly. The left-most panel lists all your notebooks, with a number in brackets denoting the number of notes they comprise. Notes in the selected notebook are displayed in the middle panel, and you can read the selected note in the panel on the right.

If you've worked with *Vim* in the past, you'll appreciate *Joplin's* modes. In the Normal mode,

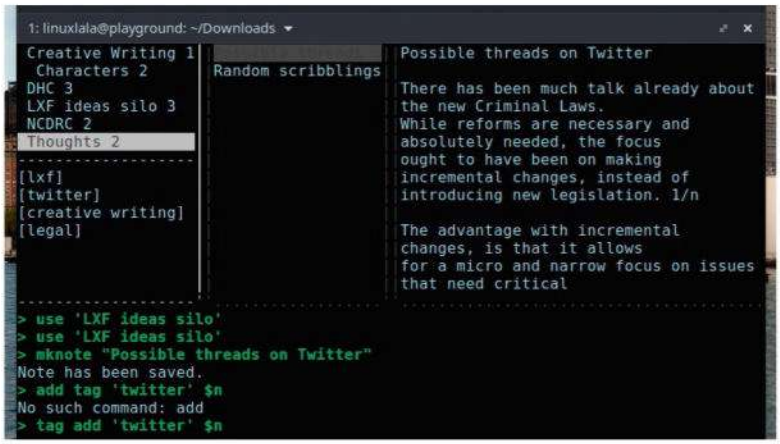
which is the default, you can press the Tab key to switch between the different panes and use the arrow keys to scroll through the notes, and hit Enter to select a note.

You can use various predefined shortcuts or keybindings to perform various operations from the Normal mode. For instance, press `mb` to create a new notebook, or `mn` to create a new note.

When you run the `mb` or `mn` shortcuts, you're asked to provide a title. This is the title of the note or notebook. You can now navigate to the note in the middle pane and hit Enter to open it in the defined editor to add content to the note.

The `mb` keybinding is a shortcut for the `:mkbook "Title"` command. You can enter Command mode by pressing `:` and then typing the command you wish to execute. Run the `:config editor "nano"` command to set *Nano* as the default editor to edit the notes.

Joplin also supports auto-completion, which is really handy and can be used to select notebooks and notes you wish to edit. The currently selected notebook or note is the one that's in focus. Some operations, such as make note, are performed within the notebook that's in focus. You can switch the focus to a different notebook by first pressing Tab to move to the notebooks pane and then using the arrow keys to select the appropriate notebook. Alternatively, you can quickly select the notebook using the Command mode. For instance, we can switch to the Thoughts notebook by running the `:use Thou<tab-completion>` command to save time.



Joplin might produce an error when you first try syncing. Make sure you've copied the complete authorisation code, then repeat the steps.

You can similarly use a note's title to open it in the configured editor with the `:edit <note-title>` command. This operation, however, is notebook-specific and not global. That is, you must first change focus to the correct notebook before you can use the `:edit` command to edit a note.

Joplin autosaves your notes every few minutes, or you can run the `:sync` command to sync your notes. To exit *Joplin*, run the `:exit` command. To delete a note or notebook, select it in the relevant pane, and then press Backspace or Delete. You're asked to confirm the operation. When deleting a notebook, all comprising notes are also deleted.

You can use the `:tag add <tag-name> <note>` command to add tags to a note in the notebook in focus.

To view all the defined shortcuts, type `:help shortcuts` and hit Enter. *Joplin* displays all the shortcuts in a new pane at the bottom of the interface, called the console pane. The console pane also lists all operations performed by *Joplin* and you can scroll through it using the up and down arrow keys.

You can press `tc` to toggle the console pane at any time and scroll through its contents. This is useful because *Joplin* often produces output that can only be accessed from the console pane, such as instructions about syncing.

It's also possible to move a note from the currently in focus notebook to another notebook. Run the `:mv <note-title> <destination-notebook>` command to move the specified note to the given notebook. The `:ren` command can similarly be used to rename your notes and notebooks.

As always, be sure to run the `:help`, `:help keymap` and `:help <command>` commands if you want to learn more about how to make the best use of *Joplin's* many handy features. LXF

QUICK TIP

As the notes are all stored in an SQLite database, *Joplin* is able to use the Full Text Search (FTS4) extension to identify notes and notebooks. You can also use operators such as tag: and -, any:, title: and so on to limit the search. Refer to the project's documentation for more details (https://joplinapp.org/help/apps/search).

» SYNCHRONISE NOTES

Joplin supports syncing your notes across various services. Each of these services is a different sync target, so you must first define the sync target you wish to use. For this tutorial, we're discussing how you can sync notes across different *Joplin* instances using Dropbox. Refer to the official documentation for the necessary steps to synchronise using different services (**https://joplinapp.org/help/apps/sync/**).

To sync notes using Dropbox, first we set the sync target. From within *Joplin*, run the `:config sync.target 7` command, then press `tc` to open the console pane so you can read the instructions. Now run the `:sync` command. *Joplin* provides a link to authorise *Joplin's* access to your Dropbox account. Open the link in your browser and click Allow. Copy the authorisation code provided by Dropbox and paste it into the *Joplin* CLI utility where prompted. When done, run the `:sync` command once again to initiate the sync process. The output in the console pane will inform you when the task is completed:

```
<output in console pane truncated for space>
Created remote items: 20.
Deleted remote items: 1. Fetched items: 26/26.
Completed: 02/07/2024 20:34 (95s)
```

You have to follow the same steps in other *Joplin* installations, such as the desktop app or on your mobile device. In the CLI utility, you have to manually run the `:sync` command to keep your notes in sync with the Dropbox version.

Part Four! Don't miss next issue, subscribe on page 16!

LINUX BASICS

Perfect your Ubuntu install settings

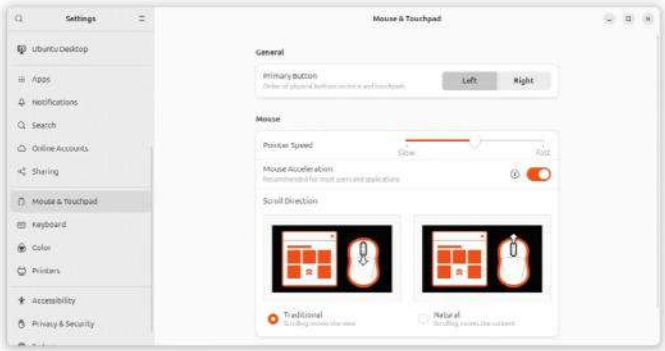
Nick Peers loves nothing better than dissecting system settings. Join him as he explores Ubuntu and Mint's respective tools.

OUR EXPERT

Nick Peers has decades of mileage when it comes to Linux. He refuses to say how much of that time has been spent productively.

Once you've grasped the fundamentals of your new Linux operating system, you'll want to start moulding it in your own image. And the good news is that all Linux distros are keen to give you the tools you need to do just that. In this issue's Linux Basics, we're looking at what configuration options are available, where to find them, and what settings to choose. Our focus remains on Ubuntu 24.04 LTS, but we'll also reveal where to look in Linux Mint, too.

In Ubuntu, everything is in the *Settings* app, accessible from both the Launcher and the System menu in the top-right corner via the cog-like button. Linux Mint users will find a handy shortcut on the Start menu. You'll also find shortcuts to specific parts of the *Settings* app from within other parts of the desktop, too – more shortly.



If you're used to the so-called natural direction of movement when you use a mouse wheel, you can configure it in Settings.

Take the grand tour

Ubuntu's *Settings* menu (see top-right) is a two-paned affair: on the left is a series of categories, from Network and Bluetooth at the top to Accessibility, Privacy & Security and System at the bottom. Click one and its settings are displayed on the right. Mint displays its various options as a series of icons, split into four categories. Click one to switch to the relevant settings.

From here we're going to largely focus on navigating Ubuntu's settings – Mint users should be able to follow along using the search tool in the top-right corner of the *Settings* app if you can't easily find the equivalent tool. For those options exclusive to Mint's *Settings* tool – such as *Firewall* – check the box *(opposite page).*

In Ubuntu, you'll notice the categories are split into five loosely related sections. The first can be described as Networking and contains two or three entries: Wi-Fi (if you have a Wi-Fi adaptor), Network and Bluetooth. All can be accessed via the System menu by clicking > next to the relevant adaptor and choosing All Networks (Wi-Fi) or Wired/Bluetooth Settings.

Networking settings

Wi-Fi enables you to manage existing saved networks as well as search for (and connect to) both visible and hidden networks. In the case of the latter, you need to know its SSID (the name that identifies it, and which is usually broadcast for people to connect to it) and what type of encryption it uses, typically WPA2 or WPA3.

Click Saved Networks to view a list of networks you've connected to now and in the past. You'll see two icons next to each entry, which are also accessible from the main screen on the network you're connected to. The QR code enables you to share the connection details (including passcode) to a phone or tablet, while the Settings button opens a separate window. The box *(page 62)* reveals the ways in which you can tweak your Wi-Fi settings and even turn your PC into a hotspot.

Select Network to access your wired Ethernet adaptor, complete with link speed, on/off toggle and access to its own Settings dialog, which works in the same way as it does for wireless connections. Beneath this are options for setting up VPNs and Proxies – the former supports OpenVPN and WireGuard protocols as well the older (and leakier) PPTP connections.

The final section – Bluetooth – enables you to configure attached Bluetooth devices. If you simply have a Bluetooth adaptor installed, your computer is already set up to receive files via Bluetooth, with transferred files placed in your **Downloads** folder.

Displays, audio and power

If the Networking section is relatively logical and straightforward, the next section is more a loose collection of vaguely related items. Displays enables you to configure your monitor – click its entry to

QUICK TIP

If you're looking for more ways to tweak your desktop, we recommend installing *Gnome Tweaks*, which you'll find in the main Ubuntu App Centre. You'll gain access to many of the same appearance-based tweaks that Mint users take for granted in their *System Settings* app.

change orientation, resolution, refresh rate and scale (useful on 4K monitors). It's also where you can configure multiple displays, making one your primary display and clicking and dragging each display to determine how you move from one to the other with the mouse. There's also a Night Light setting, where you can reduce the colour temperature at night to prevent insomnia and eye strain when working late.

Sound is where you select your input (microphone) and output (typically speakers) devices, plus configure their volume and other settings, such as speaker balance. It's also where you can choose a different sound for system alerts, as well as reduce their volume independently of your main audio volume. You'll also find a shortcut to Sound on the System menu – click > to the right of the volume slider to access it.

The Power section provides options for switching power plans from the default balanced to power saver, plus you'll find power-saving features like choosing when the display automatically blanks or goes into suspend mode, as well as an option to change what the power button does when pressed.

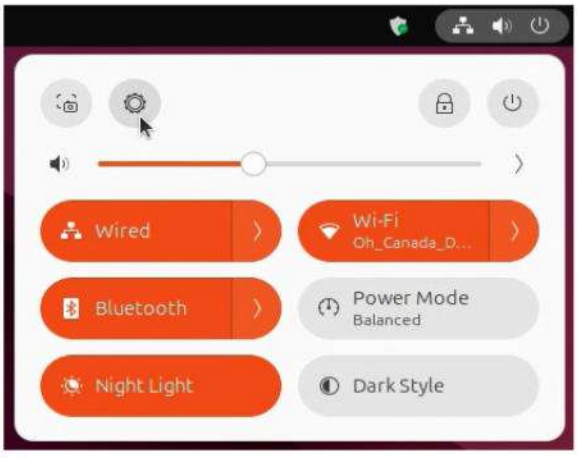
Tweak desktop options

The final three options in this section are related to the desktop, which we touched on in the first Linux Basics tutorial (**LXF315**): Multitasking is split into subsections. General contains two shortcuts – enabling Hot Corner enables you to open the Activities window by rolling the mouse into the top-left corner of the desktop, while Active Screen Edges makes it possible to resize windows by dragging them to the sides or edges of the screen (like the Snap feature in Windows). Below this are various options all linked to Ubuntu's support for multiple virtual desktops, which it dubs workspaces.

Appearance contains options for customising the desktop wallpaper as well as choosing a style (default or dark) and colour scheme (which affects the colour of desktop items). Changes are instantly applied, so you can experiment until you find something you like. The last section, Ubuntu Desktop, covers desktop behaviours *(see the first Basics tutorial for details).*

Configure apps

Settings' third category is again a hotchpotch of vaguely related items. It opens with Apps, which allows you to switch default apps for web browsing, email, calendar, music, video and photos. You can also configure autoplay here for when you insert CDs, DVDs or other removable media.



The System menu offers the fastest way into the Settings app for Ubuntu users – Mint users need only open the Start menu.

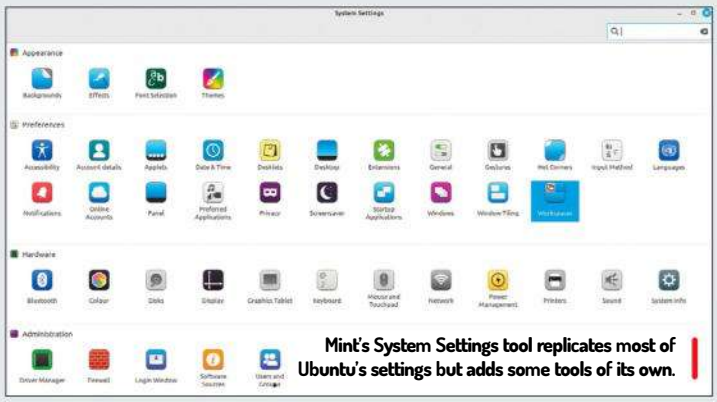
Apps is also the place to go to configure individual app permissions in relation to Linux itself. You're told if the app is sandboxed or not (Snap apps are by default), while you'll also see switches relating to permissions, from displaying notifications to allowing the app to run in the background. In most cases, the default should be fine, but if in doubt, review and tweak them from here.

Speaking of app notifications, there's a Notifications section, where you can set general settings (do not disturb, whether to show notifications on the lock screen) before choosing what type of notifications each app can display and how they're announced.

Fine-tune search

Ubuntu's universal search tool is predominantly used in both *Files* and from the Launcher, and Search is where you get to configure how it works, Start by choosing whether to include search results from various apps (those listed at the bottom of the screen). You can then enable and disable individual apps, change their sort order *(Files* heads the list by default), and configure

» DO MORE WITH MINT



Mint's System Settings tool replicates most of Ubuntu's settings but adds some tools of its own.

Mint Cinnamon's desktop ships with its own *System Settings* tool, packed with a wide array of options and shortcuts, many of which don't appear in Ubuntu's tool. Part of that is down to the fact *System Settings* includes shortcuts to other tools, such as *Disks, Driver Manager* and *Software Sources*.

However, there are exclusive features that put it ahead of Ubuntu's more limited tool. The Appearance section alone gives you far greater flexibility and control over Cinnamon, for example, with sections for effects, fonts and themes, on top of backgrounds.

Mint also excels in its Hardware section, where you'll find built-in support for tablets and styluses. Elsewhere, Mint's *Printers* tool is head and shoulders above Ubuntu's more limited offering.

Mint's settings app also flags up some exclusive tools, most notably shortcuts to Mint's built-in *Firewall* as well as a comprehensive *Login Window* component for customising all aspects of your login screen. Another highlight is *Startup Applications,* a superior alternative to Ubuntu's *Startup Applications Preferences* tool that's integrated into *System Settings* itself.

Mint also dedicates a subsection of its Start menu – Preferences – to providing shortcuts to each individual settings component, while its Administration subsection is also worth exploring.

QUICK TIP

Need help identifying a program by its package name? Try issuing the following command in the terminal: `apt-cache search firefox` **(substitute your chosen app for** `firefox`**). This searches all packages (and their descriptions) for possible matches.**

QUICK TIP

Don't panic if you fail the Device Security test under Privacy & Security. Just make sure Secure Boot is Active and you should be OK. Click the 'i' button at the top of the window or visit https://fwupd.github.io/hsi.html for more information about these tests.

which folders and drives are included as part of the search – if you have data on external drives, be sure to click Search Locations to add them to the index.

Online Accounts gives you a second chance to connect online services to your Ubuntu account, from specific providers (including Google, Nextcloud and Microsoft) to general accounts (SMTP and IMAP for email, WebDAV for calendars and contacts). The final category here – Sharing – enables you to rename your PC's name as seen on the network, plus enable media sharing, which allows the content of your **Music**, **Videos** and **Pictures** folders to be streamed across the network using any UPnP-compatible media player.

Tweak more hardware

Settings' penultimate section covers peripherals. The Mouse & Touchpad section enables you to switch the primary mouse button from left to right, alter the pointer speed, toggle mouse acceleration on and off, and switch the scrollwheel from traditional (default) to natural (reversed). A Test Settings button opens a window with tabs that allow you to test mouse buttons and scrollwheel to confirm you made the right choices.

The Keyboard section offers four options. First, Input Sources lets you switch between keyboard languages and layouts, while Input Source Switching enables you to set different input sources for different windows, instead of universally when switching using the keyboard combination of Super key – typically Windows or Apple on most keyboards – and Spacebar.

Next, you get to switch Alternate Characters and Compose keys. The Alternate Characters key is used to enter a third option on a keyboard, such as the € symbol when pressing `4`, while Compose is used to enter characters using a sequence of keystrokes.

The last option is the most useful – here you can change various system keyboard shortcuts, such as PrtScr for taking screenshots. You can even add your own custom shortcuts, which involve invoking terminal commands to run scripts, launch applications and more. In cases where you wish to launch apps, you simply need to enter the package name of the application you wish to run into the Command box, such as `libreoffice-writer` *(see Quick Tip page 61, if you're struggling to identify an app's package name).*

Colour profiles and printers

Colour profiles are designed to help monitors and printers display or print colour more accurately. In most cases, the automatic settings are fine, but should you have access to an ICC profile file for your specific display or printer, click > next to its entry and click Add Profile followed by Import File to use it.

The last option here is where you can manage your printers. Most modern printers are automatically detected as soon as they're connected and you can view their status or click the vertical ellipsis button next to the name to set printer options, search for drivers (click Printer Details) or set the printer as your default. An Add Printer button lets you search your network for potential printers to connect to.

Accessibility, privacy and security

The final section opens with various accessibility options for those who need help with various things, from seeing and hearing to using the keyboard and mouse. An Always Show Accessibility Menu switch enables you to place a shortcut in the menu bar to the left of the System menu buttons. The options are all self-explanatory and consist of simple switches or sliders. A Zoom option enables you to zoom into and out of the desktop but is somewhat unwieldy. If you accidentally get stuck after switching it on, press Alt+Win+8 to shut it off.

Make sure you spend some time exploring the various options under Privacy and Security. The Connectivity option explains why you might want to disable the connectivity checker feature, while the Automatic Screen Lock enables you to determine how long your PC waits before blanking the screen and whether that leads to an immediate screen lock.

A Location setting enables you to decide if you want to enable Automatic Device Location, which can then be used by *Mozilla* and other non-sandboxed apps (namely packages rather than Snaps). Sandboxed apps can be given access to your location data when installed or via the Apps section of *Settings*. You can review which ones have access here, plus revoke their permission if necessary.

File History & Trash enables you to control the file history list (based on the files you've opened) and whether you want to automatically delete files in the

» ADVANCED WI-FI TWEAKS

When configuring your Wi-Fi network, you may wish to learn more about your connection or make changes. To do so, click the Network Options button to the right of the connection's QR sharing button.

The dialog that pops up is a wealth of information, spread across five tabs. Details provides info about the connection's strength and speed, enabling you to reposition your PC if necessary to improve network performance. There's also details about the IP address, and an option to make the connection available to other user accounts.

The IPv4 and Security tabs can be used to change the connection, such as switching to a manual IP address or configuring how any stored password should be shared with other user accounts.

An option in the Wi-Fi section enables you to turn your Wi-Fi adaptor into a hotspot for others to use for internet access. This only works if your PC has a separate connection (typically wired) to the internet. Click Turn On Wi-Fi Hotspot, provide a network name that others will see to connect, and a strong password. Click Turn On.

A QR code appears in place of the Visible Networks list, which other devices can scan to connect through. You'll also see the Turn Off Hotspot button for when you need to end the connection.

You can turn your computer into a Wi-Fi hotspot.

trash, as well as temp files after a set period. Diagnostics enables you to decide whether you're happy to send error reports to Canonical.

The final option in this section – Device Security – performs a security audit of your hardware. Check the Quick Tip *(opposite)* if, as is likely, your PC fails the test.

System settings

The final category is itself a mishmash of various options. Region & Language and Date & Time are self-explanatory, while Users is the place to go to change your user password, add a login picture, plus configure automatic login and switch language. You can also add other users here – click the Unlock button to gain full access to available features.

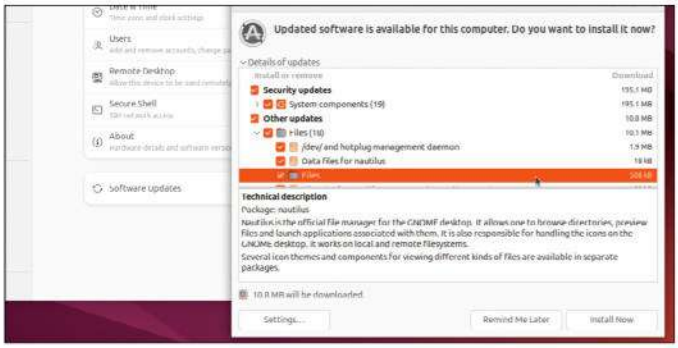
Remote Desktop and Secure Shell are options that enable you to access and control your PC from other devices. Remote Desktop offers two ways to access: Desktop Sharing is a good choice when asking someone else for help, while Remote Login is best employed when you're away from home and want the option to log into your own account when it's not being used. The Secure Shell option is for advanced users happy to log in using a terminal. You need appropriate client software on your other device for access – any RDP-compatible client such as *Remmina* (**https://remmina.org**) will do for Remote Desktop connections, while secure shell connections can be made from any terminal on a remote Linux-powered PC or *Putty* (**https://putty.org**) on Windows PCs.

The final two options enable you to view limited information about your PC's hardware specs, including processor, RAM and total hard drive capacity, while Software Updates gives you the opportunity to manually check for and install updates – see the walkthrough *(below)* for full details. LXF

KEEP YOUR SYSTEM UP TO DATE

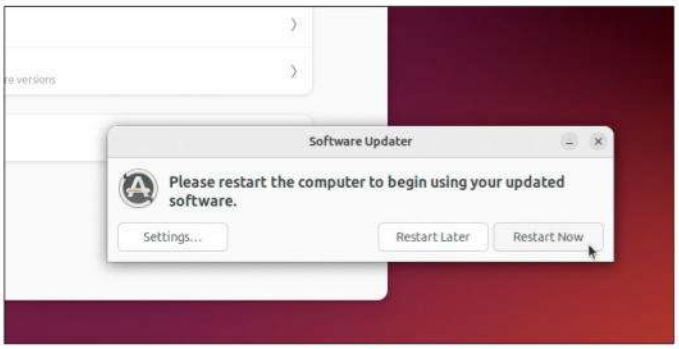
1 Check for system updates

You can manually check for updates via *Software Updater,* which can be opened from the Launcher or accessed via *Settings* – navigate to System and click Software Updates. If necessary, click > next to Details Of Updates to reveal the list of available updates, along with an estimated download size – both an overall figure and for each individual package.



2 Review updates

To find out more about what's being downloaded, expand each section by clicking the > next to it, then click on an entry. You're given its package name, a description and a list of changes (scroll down for these). If for any reason you decide against installing a component, simply remove the tick next to it – we don't recommend skipping security updates, however.



3 Install updates

Click Install Now and Ubuntu installs the selected updates – enter your user password when prompted. *Software Updater* exclusively installs package updates – the same that are downloaded and installed when you issue the `sudo apt update` command followed by `sudo apt upgrade` in the terminal. After installing, you may be asked to reboot.

4 Configure updates

Click Settings in *Software Updater* and you're whisked off to the Updates tab of the related *Software & Updates* tool. One recommended setting here is to verify security updates are set to download and install automatically in the background. For more on installing and updating software in Linux, check out last month's Linux Basics tutorial *(see page 64 to order an issue).*

BACK ISSUES » MISSED ONE?

ISSUE 317
August 2024

Product code:
LXFDB0317

In the magazine
Don't consign your old machines to the scrap heap – they can still be put to work with Linux. And join us as we dive deeper into Linux to explore virtualisation, containerisation and virtual filesystems. We also have tutorials on managing software and files, emulating VDUs, creating old-school pixel art, and more. Plus we have our usual news, views and in-depth reviews.

ISSUE 316
July 2024

Product code:
LXFDB0316

In the magazine
Transform your desktop with hot KDE Plasma, and join us as we start our journey inside Linux to discover how it works. Plus, choose the best text editor for your writing projects, find out about the phone firm making an environmental impact, build your own weather machine, get back to basics with the Linux terminal, keep your video calls private, and enjoy a wealth of reviews, news and views.

ISSUE 315
June 2024

Product code:
LXFDB0315

In the magazine
Discover how to upgrade to Ubuntu 24.04 LTS and join us on a journey through the world's most popular distro's 20-year history. And don't miss our *Roundup* of hacker distros, an in-depth look at the impending Epochalypse, detailed tutorials on live-streaming audio via the terminal, setting up Gnome, editing images in *Krita,* and archiving C64 tapes, and plenty more besides.

ISSUE 314
May 2024

Product code:
LXFDB0314

In the magazine
Join us as we explore Linux's infiltration of Windows, and discover how you can also get Windows apps running in Linux as we uncork the latest *Wine* release. And let's not forget a *Roundup* of time trackers, tutorials on vulnerability audits, setting up a home ebook server, streaming games, and more, along with distro reviews, news, and packed Pi and *Adminsteria* sections.

ISSUE 313
April 2024

Product code:
LXFDB0313

In the magazine
Discover how to use the ultimate hacker's toolkit, staying out of trouble while doing so. And join us as we take the Puppy Linux developer's new distro for a run and explore its container features. Plus, we have a *Roundup* of retro-gaming distros, an in-depth look at filesystems, tutorials on adding plug-ins to our **LXF** shell and getting more from your VMs, plus news, reviews and oodles more.

ISSUE 312
March 2024

Product code:
LXFDB0312

In the magazine
Blast off into the future with a look at the five best next-gen distros, and discover whether Raspberry Pi or Orange Pi is the best SBC for you. Plus, learn how to rescue retro media, add NPCs to your own point-and-click adventure, emulate an analogue computer, and lots more. We've also squeezed in hardware and distro reviews, a password manager *Roundup*, news and more.

To order, visit www.magazinesdirect.com

Select **Single Issues** from the tab menu, then select **Linux Format.**

Or call the back issues hotline on **0330 333 1113**
or **+44 (0)330 333 1113** for overseas orders.

Quote the product code shown above and have your credit or debit card details ready.

AUDACITY

Credit: www.audacityteam.org

How to become a podcasting pro

Podcasts are the modern way to get the word out about issues you care about. **Michael Reed** shows you how to record, edit and publish your own.

OUR EXPERT

Michael Reed is an old-school Linux man who compiled the kernel to free up memory on his 486 PC.

The internet supports a dizzying mixture of formats and platforms, but of all of them, podcasting perhaps presents the best balance of being informative, entertaining and intimate. Also in its favour, it's a medium that you can enjoy while doing something else, such as commuting or washing up.

The format of a podcast is most commonly audio only and the subject is usually tied to a specific topic, such as gaming or travel. Many of the most successful ones come out weekly and hover between 15 minutes and an hour in length. Podcasts are extremely easy to make, once you learn the basics and get into a routine.

Podcasting preliminaries

Let's get down to the basics of what we're making. A podcast is an audio file that you normally distribute in the MP3 format. You make the podcast by recording it into your computer through a microphone, which gives you an audio file in WAV format. Typically, you do some editing in audio software before exporting the final product. Then you upload the finished MP3 file to a podcast broadcasting service so people can listen to it online or download it to a portable device.

Recording the audio and editing it is relatively simple. However, making a successful podcast that garners some regular listeners is a bit more tricky. First, choose an overall subject that you'd like to cover. Look for a subject that isn't already swamped in the podcast market. Ironically, you're more likely to have some success by choosing a niche subject. For example, when you're just getting the subject established, a subject like 'gaming' is less likely to be successful than specific 'open source Linux gaming'.

Choose something you're an expert on and are interested in. When trying to get a listener base established, you need to be able to keep up the enthusiasm (for you and them). In addition, people want to listen to someone who knows a lot about the topic, which plays into the niche-centric nature of podcasting. Maybe you're quite the expert on installing Linux on vintage hardware or mobile devices? Maybe you're always on top of the latest gaming releases on Linux?

If you prefer, you might like to summarise your points using the nested bullet point list in a word processor.

- Game name
- Developer
- Development
  - 12 months
  - first alpha release
  - Unity game engine
- The positives:
  - Good graphics / visual design
  - Interesting setting (cyberpunk)
  - Some good mechanics
    - resource gathering
    - clever AI
- The negatives:
  - Graphics bugs
  - Not many units / buildings
- Conclusion:
  - A lot of potential / Definitely worth following
  - Developer website

Planning – do some!

One of the easiest mistakes to make is a lack of planning. Completely rambling podcasts tend to struggle to find listeners (unless there's a celebrity attached). When you hear a podcaster who makes it seem easy and natural, odds are they are making use of a plan that they wrote in advance.

Make notes on a constant basis. So, every time you think of something you'd like to mention, add it to your podcast notes file. Using a news feed aggregator such as Feedly (**https://feedly.com**) is a good way of mining for ideas. We recommend using an online service like Google Docs rather than a file-based word processor, as you can add to your notes every time you come up with an idea and keep your notes where you can see them while recording the podcast.

There are two levels to your preproduction notes: the list of segments (the name for the sections in a episode) and the bullet points of what you want to cover in each section.

Create segments

The overall structure of the podcast is the first thing to consider when making a plan. A reasonable layout would be something like: introduction (one minute), latest news (three minutes), three reviews (three minutes each), closing thoughts (three minutes). That gives a total length of 16 minutes.

A reasonably short podcast means you have to remain disciplined to fit the segments into the allotted length, which tends to improve the overall quality. Another thing that makes podcasting special is that you can vary the length from one episode to the next –

for example, an interview is a great idea for a segment, but you might not be able to obtain one every week.

The introduction is a short segment in which you introduce yourself and other participants, summarise what the podcast is about and what topics you'll be looking at in this episode.

The closing thoughts could include some personal details such as what games you are playing at the moment and anything interesting going on in your personal life. It's also a good idea to add any news about the podcast itself, including the services that it's available on and when to expect the next episode.

Case study: a review

Let's say that the subject of your podcast is Linux gaming. On a given episode, it might be that there are three interesting games you want to cover.

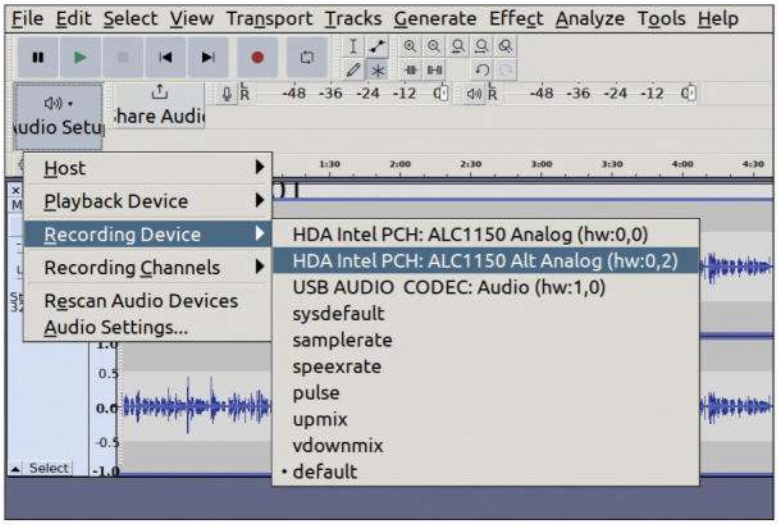
The first is a strategy game. It's in early alpha but considered to be in a playable state. The excellent graphics and music complement the setting (cyberpunk futurist) with a great atmosphere. The enemy AI is extremely challenging and there's a number of interesting gameplay mechanics. You did however, have some problems with unit selection within the user interface. As it stands, there is a paucity of unit and building types. Your conclusion is that it's still at an early stage, but definitely worth keeping an eye on to see what future developments bring.

Here's how you might summarise that in bullet point prompts that keep you on track while talking: game name; developer; development (12 months, first alpha release, Unity game engine); positives (good graphics/visual design, interesting cyberpunk setting, some good mechanics – resource gathering, clever AI); negatives (graphics bugs, not many units/buildings); conclusion (worth following); developer site and link.

A list like that, available on a screen you can see while recording, should keep you on track and prevent some of the stammering and rambling that can make a podcast less interesting to the listener.

Interview segments

Interviews are a great example of an engaging segment that make people want to listen to your podcast, and the interviewee and their followers usually want to link to and promote that episode of your podcast. Free publicity!



Audacity's Audio Setup pop-up menu (added in the 3.x series) allows for quick setup of your audio ins and outs.

There are many ways of recording interviews, and there is no reason to lock yourself into your main recording workflow when gathering interview audio. For example, many of the instant messaging clients, such as *Telegram,* have a built-in recording feature. Most phones can be persuaded to record a call, too. Check to see which method of interviewing is most convenient for the interview subject. Do a quick test of the setup with a friend in advance to make sure the recording works and the files are usable.

Again, planning is key when interviewing. Have questions written up before making the call and bear in mind that interview subjects can give replies that range from terse to verbose, so your questions can produce anything from five minutes to half an hour of audio.

Once you have the audio in a format like WAV or MP3, drag it into your *Audacity* project, move it to the right location and edit it.

QUICK TIP

While working on a podcast, save the project (File >Save Project > Save Project) rather than export the audio, to keep the project in an editable state.

Using Audacity

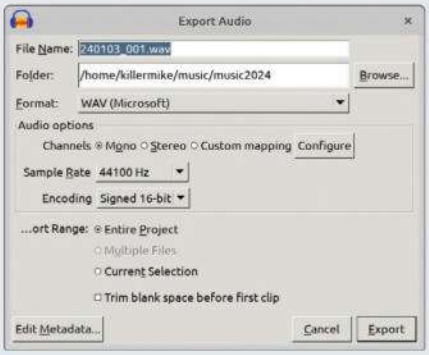
The first stage is making sure that *Audacity* is installed. In our experience, the version of *Audacity* featured in the repository of most distros is out of date, and that's a shame, given that *Audacity* is updated at a fairly

» PUBLISHING

When happy with your editing, it's time to export the file ready for publishing. Export a stereo audio file (File > Export Audio). Most services can accept an uncompressed WAV file, for maximum quality, but MP3s should suffice.

Services such as Soundcloud and YouTube are an easy, free way to get your content out there, but dedicated podcast services integrate better with podcast listening services and apps. Podbean, Spotify, Sounder.fm and RedCircle offer decent free plans.

Most podcasting services operate on a freemium model. Go to each of the pricing pages to compare the features between the free and paid tiers. The amount of storage space might be significant if you intend to create longer podcasts. Check the list of directories that the service publishes to. Analytics are important to check your progress and fine-tune your podcast when getting started. We spotted that one service only hosted the podcast for 90 days on the free plan and another had a limit of three episodes on its free plan, both restrictions that would spoil things for a podcaster who's just getting started.



Audacity's export dialog. Get used to seeing this, because it's the final stage of the process before actually publishing on a service.

Having split an audio clip and resized it, it's time to push the two clips together for a seamless edit.

decent pace. You've got a few choices. The most prominent option on the official *Audacity* website (**www.audacityteam.org**) is the AppImage. This has all the advantages of being a self-contained archive that doesn't install itself. You simply have to download it, make it executable and double-click on it to run it.

The downside of an AppImage is it never fully integrates into your system, so it's worth searching the Flathub and Snap repos. Having installed the Snap version, we were prompted to grant permission to the audio hardware with a command-line sequence.

QUICK TIP

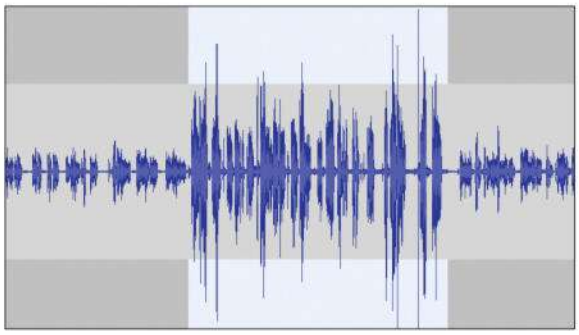
Hitting hotkeys while recording is a useful technique, but keypresses are picked up by a microphone. Increase the distance between microphone and keyboard or simply press the keys as softly as you can.

Recording the audio

We'll assume that you've used one of the available methods to install *Audacity 3.x* and that you have a suitable microphone/audio interface combination.

Plug your microphone in and try a test recording. Make sure your microphone device is selected in the menu that pops up when you click on the Audio Setup icon, then click on the Record icon and try recording some speech. If you have a 24-bit audio interface, you have a lot of leeway in terms of recording on a low volume, then increasing the volume afterwards without a loss of quality. However, try to get the peaks (highest parts) of the audio to about halfway up the waveform track. Even with the best equipment in the world, you're always going to bring up the noise floor (random background electrical noise) when you increase the volume of an extremely weak signal. When working with digital audio, however, the golden rule is never to record at above the maximum level of your equipment and software (labelled 0dB) as it sounds awful and there is no way of repairing the damage.

Once you have confirmed that you can record at an acceptable volume, it's time to record your first podcast. Make sure your notes are visible from where you are sitting. Take full advantage of the fact that you're working in the digital audio age by stopping and starting whenever you need – to re-examine your notes catch your breath, or have another go at saying something succinctly.



Only the middle part of this waveform has been compressed. Notice how it fills up more of the available range.

Record workflow

Audacity is well set up for podcasting in terms of workflow. If you stop a recording that is in progress, the position cursor also stops. When you restart the recording, a new audio section is created that begins precisely where the old one ended. This makes it easy to see at a glance where you have restarted a section.

Let's say that you made a flub in how you said something: stop the recording (click on Stop or press Space), then restart it (click on Record or press R). Say what you wanted to say again, hopefully without the mistake. You can edit out the mistake now or leave it until you've finished recording. It's your podcast – if you have to keep reattempting part of what you're trying to say several times before you get it right, that's fine. All that matters is the end product, and edits made in *Audacity* will seem seamless to the listener.

Create markers to show the start of each segment (Ctrl+M). You can name the markers by clicking on their label. The first time you do this, *Audacity* creates a Marker Track, which sits beneath the main audio track.

The final edit

Once you have recorded your podcast from start to finish, it is time to begin the editing process, and this is where you can make the whole production shine.

When dealing with longish audio material (such as an unedited podcast), *Audacity* zooms out to show you the entire audio file. However, it's usually better to zoom in to a level where you can clearly see the individual words by using the + icon or hovering over a section of audio and using the mouse wheel and Ctrl.

Time to remove the extraneous material (the technical term for waffling). Click once on the waveform on the area you need to cut. Split the clip at this point (Split Clip in right-click menu or Ctrl+I). You can now drag the start or end of the clip to remove the unwanted material. You can move the clip itself by dragging the clip by its title bar. Drag the clips together to eliminate gaps.

This process takes advantage of the non-destructive nature of *Audacity*. Doing it this way is a tiny bit of extra work with the mouse (compared to drag-selecting an area and pressing Delete), but as the underlying file is never altered, it means you can go back and drag the start or end of a clip to reveal any material that you removed.

You can also quickly delete audio by highlighting it and pressing Delete. However, you can't undo this later on in the editing process.

When deleting spoken audio in the waveform view, always cut between the words. Apart from the fact that you're unlikely to want to cut words in half, if you cut audio when it is not crossing the centre line (vertically), you get an audible popping sound.

Compress the audio

Natural speech varies quite a bit in volume, and podcast recordings can sound a bit weak when you first play them back. If you listen to a radio DJ, you'll

notice that the voice is always constant and punchy sounding. The tool to automatically even out the inconsistencies in audio level is called compression.

To apply compression, select all of the audio (Ctrl+A) and navigate to the compression plugin (Effect > Volume And Compression > Compression). You're going to have to play around with the values a bit, because a lot depends on how variable the volume of your recording is and how much it needs boosting in overall volume. Settings that are too harsh lead to a distorted sound. The most important settings are Threshold (the level at which the compression starts to take effect) and Ratio (how much the volume is affected by the compression). In our case, we found that settings of Threshold -12dB and Compression 8:1 caused the audio to fill the full waveform to a greater extent, resulting in a thicker, louder sound.

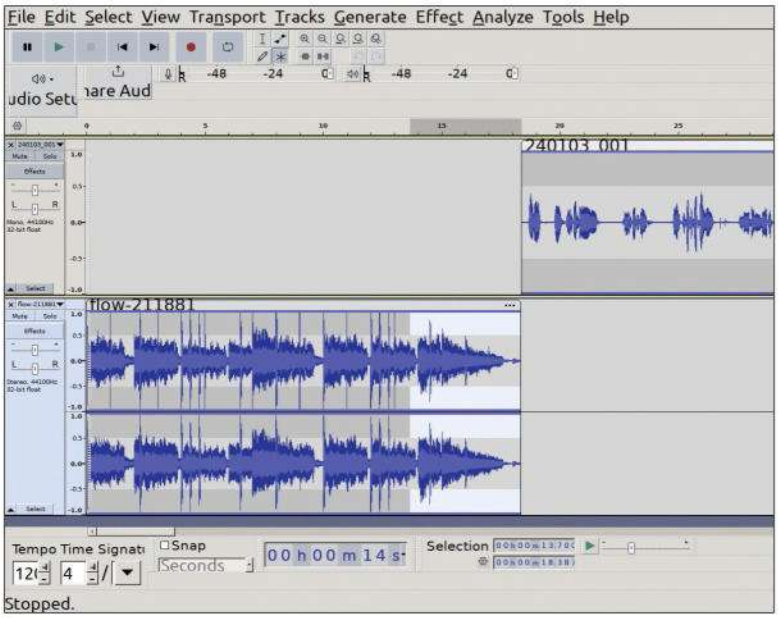
Add some music

Having an introductory piece of music on your podcast can help with establishing your brand and creating a familiar atmosphere for the listener. Always check the licence for any music you intend to use. Thankfully, there are loads of music sites on the internet that offer music with a permissive licence for just this reason. Pixabay's music section (**https://pixabay.com/music**), SoundCloud (**https://soundcloud.com**), Free Music Archive (**https://freemusicarchive.org**) and Kevin MacLeod's website (**https://incompetech.com**) are all great sources.

If you drag a music file to the *Audacity* project window, it is placed on its own track. If you want it to be used purely as an introduction to your podcast, you have to remove the overlap.

Trim the audio track by hovering over the right-hand edge and dragging it to the left to reduce its length. Use the ruler at the top of the waveform area as a guide for length. About 15 seconds is probably about right for your intro music.

If you have a speech track that's split into multiple clips due to editing, you need to move all of the clips on a track at the same time. Each track has a header section, on the left-hand side of the window, and this features a Select button. Press this to select all of the clips on that track. You can then move all of the clips on that track by clicking and dragging the title bar on any of them. Use this to make room for the intro music.



Adding some introduction music (with a fade out) on its own track and before the speech starts.

Fade the very end of the intro music by dragging over the final seconds of the music and then selecting Effect > Fading > Fade Out in the main menu. It might also be worth putting music between the segments to add another professional touch.

Get podcasting!

The final finishing touch is to create some artwork to go with your podcast, as most services require at least one image to use as an icon. Either create something yourself with a photo or use a service such as Pexels (**www.pexels.com/search/podcasts**) or Shutterstock (**www.shutterstock.com/search/podcast**) to obtain a royalty-free image. Once you're happy with the episode you've created, export it and upload it to a podcasting host *(see box, page 67)*. Happy podcasting!

QUICK TIP

Due to a long-standing *Audacity* bug, the horizontal scroll bar sometimes disappears. If it's not at the bottom of the screen, resize the main window to make it come back.

» PURCHASING A MICROPHONE

In theory, you can start recording podcasts with nothing more than a mobile phone and recording app, or a laptop's built-in microphone, and setups like that are OK for getting started.

To get to a more professional level, the two basic choices you have are an external USB audio interface with a condenser microphone or an all-in-one USB microphone.

In the case of an external interface, make sure you buy an interface that can support the microphone that you purchase. To support a condenser microphone, this means that the interface needs an XLR connector and can supply 48 volts of phantom power. See **LXF315** for our complete guide to USB audio interfaces.

If you need to record two people at once, it's best to buy two mics and plug them into the separate inputs of your interface. Once recorded, split the stereo track into two mono tracks, so you edit them and adjust their volumes separately (Track > Split Stereo Track).

A USB microphone is effectively a USB audio interface, a microphone preamp and a microphone all built into a single device. They are extremely quick to set up and present as a separate device under the Linux audio standards such as ALSA and JACK.

Turn the page for reviews of four highly recommended options.

Whatever type of microphone you pick, make sure to also get some sort of stand. Cheap desktop stands are fine if you don't touch the table around them while recording. Bendy ones that can clip on to a desk or a bookshelf are worth considering.

Neat Microphones King Bee II

SPECS

Plug: XLR
Freq: 16Hz-20kHz
Type: Condenser
Polar: Cardioid
Range: 134dB @ 2.5k ohms
Size: 216x76mm, 1.12kg

Delivering a rich, realistic sound, this is perfect for vocals and instruments. At £160, it's reasonably priced and even includes a shock mount and pop filter for added value.

The King Bee II uses a 36mm condenser capsule. It features a wide frequency response range of 16Hz-20kHz, exceeding that of many competing condensers. That wide frequency response range allows it to deliver exceptional clarity and detail for lifelike audio capture.

The microphone comes with a single cardioid polar pattern. For most people, this is all they need. Cardioid is tuned to pick up sounds occurring directly in front of the capsule, reducing the sound from the back and sides. This is a great fit for single-source recording, like when you're recording solo, but if you're interested in face-to-face interviews or a multihost podcast, a multi-pattern mic is a better fit.

The King Bee II connects over XLR instead of USB. You need a USB audio interface capable of delivering 48V of phantom power to drive the mic and send its signal through to your PC. That's an added cost.

It features an exceptionally low amount of self-noise (6dB), so there's nearly no audible hiss generated by the mic itself. In combination with the frequency response range and sound character of the mic capsule, this helps ensure that what you hear on the recorded track is exactly what went into it, without the mic getting in the way. Cheaper interfaces often have low-quality components that create a quiet hissing sound. The included pop filter and shock mount are custom-made to match the mic. ***Christopher Coke***

All of the Neat microphones offer a Bee-like design style.

VERDICT

DEVELOPER: N Neat
WEB: www.turtlebeach.com **PRICE:** £160

» **Rating 8/10**

Elgato Wave DX

SPECS

Plug: XLR
Freq: 50Hz-15kHz
Type: Condenser
Polar: Cardioid
Imp: 600 ohms
Size: 53x53x145mm, 440g

The Elgato DX is the definition of minimalism. It's an end-address mic, so its rectangular body shotguns directly at your mouth. Since it uses an XLR connection without phantom power, there's no lighting, but hidden inside its hardened steel shell is a dynamic microphone capsule selected in partnership with Lewitt Audio, a respected brand.

Alongside the capsule, Elgato has implemented a built-in shock mount and pop filter. The pop filter in particular is quite good. The shock mount lets through a bit too much noise, however, so you may still want to look for another solution there.

The chassis is peppered with a 360-degree grille except for a strip on the right side. You still need to talk into the end of the microphone for the best quality, but you don't have to be quite so spot-on to be heard well.

The only other features are the rear XLR port and the mounting arm to attach to a stand – not included.

The Elgato Wave DX offers a full-bodied, broadcast-style sound signature. As a dynamic microphone, its capsule applies a bit of crunch to the voice, giving it radio-like character while still maintaining high resolution, natural-sounding vocal capture. The sound is warm, so your voice has presence, but it doesn't sound dull or over-compressed.

The integrated pop filter also works very well. We're usually disappointed by built-in filters, but the Wave DX does a great job of blocking plosives. Even speaking very close to the mic, we weren't able to get it to distort with any kind of plosive test using normal speech. Peter's peck of pickled peppers is safe with this one. The Wave DX is a great microphone in a lot of ways. It sounds rich and full, making it perfect for vocal capture. ***Christopher Coke***

An elegant, no-nonsense design from Elgato.

VERDICT

DEVELOPER: Elgato
WEB: www.elgato.com **PRICE:** £100

» **Rating 8/10**

Blue Yeti X

SPECS

Plug: USB
Freq: 20Hz-20kHz
Type: Condenser
Polar: Cardioid, omni, bi, stereo
Imp. 16 ohms
Size: 110x122x 28mm, 519g (1.28kg with stand)

This sturdy, handsome mic is immediately at home among pro audio equipment. The satisfyingly weighty base has a chrome finish, while at the front, the smart dial controls everything from gain and headphone volume to blend and mute.

MicroUSB and 3.5mm (for monitoring) inputs are at the bottom of either side of the mount. At the rear, a single button switches between cardioid, stereo, figure-8 and omnidirectional polar patterns.

In audio terms, the Yeti X's biggest improvement over the original Yeti is in its improved recording quality, up from 48kHz/16-bit to 48kHz/24-bit. What that means is that the Yeti X has a broader language for different loudness levels and is able to capture a much wider dynamic range than the original model.

Given our proclivity for slapping a compressor on everything and then uploading it to YouTube, which itself compresses audio and video, you might wonder what benefit that brings. And it's true that most audio you record won't stay gloriously uncompressed and whispery. But just as headphones designed for hi-res audio still seem to sound better at CD quality, the Yeti X has a mildly fuller sound even if compressed slightly.

Most of the time, we used the cardioid polar pattern, designed to pick up whatever's directly in front of the mic and the best option for recording a lone voice. Omnidirectional mode was great for getting some

Sturdy is one word you could use to describe the Yeti.

room noise with an instrument recording or capturing group speech, while the bidirectional mode is fit for capturing two people. All these modes sound fantastic.

The Yeti X brings welcome updates to its controls and light customisation for the aesthetically minded, but owners of the already-excellent standard Yeti don't have much incentive to upgrade. ***Phil Iwaniuk***

VERDICT

DEVELOPER: Logitech
WEB: www.logitechg.com **PRICE:** £125

» Rating 9/10

AKG Ara

SPECS

Plug: USB 2.0
Freq: 20Hz-20kHz
Type: Condenser
Polar: Cardioid, front & back
Imp: 13 ohms
Size: 182x114x 221mm, 665g

The AKG Ara is a mighty fine USB mic. Anyone trying to get excellent audio for not a lot of money should consider it. It's easy to use, as it's plug-and-play, sounds good, especially for the price, and has enough functionality for most people.

AKG mics have been well regarded long before USB mics were around, so it's no wonder that the AKG Ara is well built. With a wide, tapered mic body as well as a round base and articulating stand, the Ara looks like it would fit on a newscaster's desk in the 1960s. Of course, there are some modern touches. To start, two control dials protrude from the front. The top one lets you select between its two mic patterns – front and front and back – while the bottom controls head volume. You can press the lower dial to mute the mic.

While the stand doesn't have any swivel, it's ergonomic enough for most people's use. To start, it's light enough to pick up and adjust. Even better, you can tilt it almost 360 degrees, as it goes all the way back or forward until the head of the mic hits the stand. The base also unscrews so you can mount it on a stand.

The mic itself has a nice, full sound with plenty of volume. In fact, it's almost too loud. It also produces a somewhat bottom-heavy sound when on the front polar pattern. But, if you can adjust the low end via EQ, the mids are nice and rich, and the highs are decently clear. What you get overall is a rich but balanced sound.

Offering a classical design, it's somewhat let down by the plastic look and those cheap controls.

Regarding the polar patterns, the Ara has just two, front (or cardioid), and front and back, which is a figure of eight pattern. While we would have preferred omnidirectional for the second one, the front and back pattern does well for picking up both sides of the mic equally. That makes it ideal for conversations between two people. ***James Holland*** LXF

VERDICT

DEVELOPER: AKG **WEB:** https://uk.akg.com/microphones **PRICE:** £50

» Rating 9/10

STEAM

Credit: www.steampowered.com

Faster ray tracing on your Steam Deck

Through the wonders of Proton, Valve Software, AMD APUs and, least of all, **Neil Mohr**, you can enjoy ray tracing on the go.

OUR EXPERT

Neil Mohr can't pry the Steam Deck from his nine-year-old's hands since he discovered the joys of *Portal*.

It's not something we've written about and it wasn't even part of the launch marketing, but the Steam Deck is capable of ray tracing. We should, however, manage your expectations – though where those expectations might have been in the first place is anyone's guess – because ray tracing is phenomenally demanding on hardware and the Steam Deck wasn't tuned to run ray-tracing titles, so frame rates are going to struggle to make double figures. But with the right tweaking, it can be done, as we'll see.

Our main focus is the Nvidia-released *Portal RTX*. This is a rerelease of the classic 2007 Valve Software game with native Nvidia RTX ray tracing implemented, so it all looks shiny. There's a couple of things there that should stop you in your tracks, such as: this is for Nvidia hardware and it's using RTX ray tracing for Windows. How does that work on an AMD-powered handheld running on Linux?

The Steam Deck uses the *Wine*-based Proton translation layer to switch Windows calls to Linux. For graphics and gaming, that means DirectX calls are sent to the Vulkan library, which supports native ray-tracing hardware. Inside the Steam Deck is the AMD APU with eight integrated RDNA 2.0 compute units for its GPU.

So, as we'll see, when you install *Portal RTX*, despite it claiming no there's Steam Deck support, when you force it to use Proton, it happily runs. We can then use some clever modding from NightSightProductions that implements optimised settings and gets ray-traced *Portal* up and running at 30fps.

Ooof, that is 11 frames per second of pure ray-tracing Cyberpunk 2077 beauty on the Steam Deck.

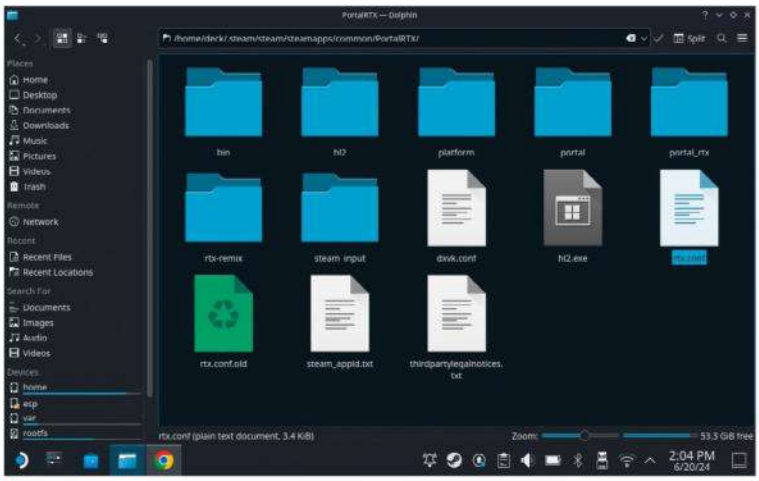
Through the Portal

You need to own the original *Portal*. It was released in 2007 and there's a good chance you already own it. If not, it's usually £8.50 in the UK, though you can find it in most Steam sales for 85p – annoyingly, the Steam Summer Sale finished on 11th July, and its next generic gaming sale is in November.

With the full *Portal* game in your gaming library and downloaded, you can search for and download *Portal RTX*. Make sure you're not searching on Steam Deck-compatible titles or you won't find it.

Switch to Steam Deck Desktop mode. Ensure you're not in Family mode and select Power > Desktop. It's always easier to use a USB-C dock with a keyboard and mouse at this point. Open *Firefox* or *Chrome* so you can grab the mod files.

Due to updated system files, the latest mod required is just a new config file that tweaks the *Portal* config settings. Previously you



Once you have the mod files, just copy and paste the one you want in Desktop mode.

QUICK TIP

If you send your children to bed early, this gives you time to play on your Steam Deck in the evenings. Just make sure there's some charge left in it...

CREDIT: CD Projekt

Here you can see the standard and ray-traced versions in action.

needed to copy optimised DXKV files, but that's no longer required. Grab the 18kb config files from **https://bit.ly/lxf318mod** or search ModDB manually.

Extract this and you'll find three preset quality levels: choose an NIS file if you prefer a pixel feel or the TAA file for a smoothed rendering. We're going with Balanced, as this provides a playable 30fps frame rate. Copy the **config.rtx** file and paste it into this folder, replacing the existing RTX file (feel free to rename the original with **.bak** first, and ensure you have hidden files visible): **home/deck/.steam/steam/steamapps/common/PortalRTX**. You can log out back to the normal Steam Deck interface.

Back in Steam, select Portal RTX > Settings cog > Properties. Under General > Launch Options, add `-NOGAMEPADUI` to the Launch Command text box. This ensures the controls work, though we did still find the mouse to be offset a little on the menus. Under Compatibility, tick Force The Use Of A Specific Steam Play Compatibility Tool, and select Proton 9.x. It's also recommended to press the ... Quick Settings button, then Performance > Manual Clock, and select the full 1,600MHz. Don't forget to put this back unless it's winter and you need to warm your house up.

You should now be enjoying ray-traced Nvidia-powered *Portal RTX* on a lowly Steam Deck. How does that work? Well, the game is actually running at 864x486 resolution, with secondary ray bounces disabled in the RTX Remix developer menu (press Alt+x to access this if you're interested), which seems to be the secret to maintaining a decent frame rate on the Steam Deck APU.

Portal RTX uses four raw bounces on Ultra settings, which is enough to bring any modern GPU to its knees at high resolution, so disabling secondary bounces and lowering the res takes the strain off that poor Deck.

Despite the chunky resolution settings used here and the fact that many game scenes look a lot darker than normal, the end result looks like a playable and very pretty rendition of the game.

Ray-tracing titles

If this has piqued your interest in how ray-tracing gaming stands on Linux (but do remember, most of these are Windows games running through Proton), there's a number of titles worth taking a look at. See our boxout if you're not a Steam Deck owner to find out what hardware you need in order to try ray tracing.

The first Steam Deck-supported ray-tracing game was *Doom Eternal,* enabled in March 2023 in the beta release of Steam OS. If you select the 720p resolution with medium settings and ray tracing engaged, you can enjoy a decidedly bloody experience at 30fps.

Another strong title to pop on your Deck is *Crysis 2 Remastered.* This not only answers the question of whether it can play *Crysis* (yes), but also re-implements the classic 2011 game with ray-tracing options. It's the same story: select the native 720p resolution with medium settings, and you'll happily enjoy 30fps.

The final title we'll mention is the most recent, the most impressive, but also the worst performing. *Cyberpunk 2077* actually has a default Steam Deck performance setting that offers a cool 50fps with no ray tracing. You can experience the ray tracing, but don't expect speeds much higher than 10fps.

It's amazing to see ray tracing landing on Linux – and it's impressive that it's running on the Steam Deck at all. While it might be more of a novelty and you could argue that it's a distraction from implementing other game features, the fact that this enables ray tracing for desktop systems as well can only be good. LXF

» DESKTOP RAY TRACING

Just because we're talking about the Steam Deck doesn't mean desktop users need to be left out. However, the Steam Deck is almost a console platform, so it's easier to tailor performance for playable results; even *CyberPunk* comes with a Steam Deck profile.

For desktop users, you need to know which GPUs support ray tracing. For Nvidia, these are the RTX models from the GeForce 2060 series onwards. For AMD, it's the RX 6000 series onwards, using at least RDNA 2 architecture. That last bit is important, as the Steam Deck's APU has eight RDNA 2 compute units, which is why it can do ray tracing. Standard PC desktops can also run these APUs – such as the Ryzen 5 8600G reviewed in **LXF314**. To have any luck, you need at least an AMD Ryzen 7000G APU that uses RDNA 2, or the more recent Ryzen 8000 APUs that use RDNA 3 compute units.

Intel's discrete Arc GPUs support ray tracing and we believe its latest 14th-gen Meteor Lake integrated graphics will, too, as it uses the same Arc Xe engine, but we have no idea about performance.

LIBREOFFICE

Credit: www.libreoffice.org

Open legacy Microsoft documents in Linux

Say goodbye to proprietary formats, as **Nate Drake** helps give your old Microsoft files a new lease of life on Linux.

OUR EXPERT

Nate Drake broke out of his cubicle at Apple eight years ago to become a freelance journalist specialising in cybersecurity and retro tech.

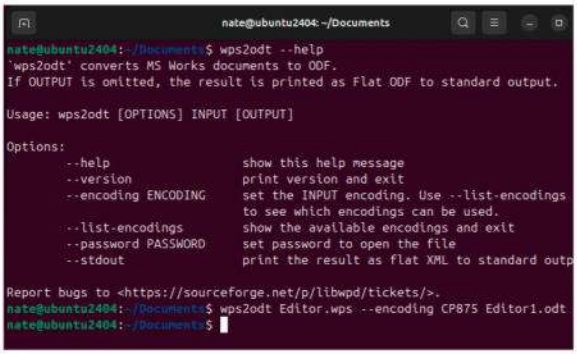
In 1989, the father of the internet, Tim Berners-Lee, made a momentous decision. To give HTML and the world's first web browser to the world freely. Since then, there's been much speculation about what would have happened if he'd tried to patent his code or web browser. The most likely scenario would be a fragmented internet behind paywalls, which not even Google could access.

As we continue to reap the benefits of a relatively open internet, it's easy to forget the main advantages of embracing Linux and royalty-free code. Even if an open source project is abandoned, as recently happened to *Neofetch,* the source code is freely available for others to maintain and fork.

Sadly, the gods of Microsoft haven't always read this particular memo. For decades, the company has released a number of closed-source products using proprietary formats.

Chief among these was the *Microsoft Works* productivity suite, which contained a basic word processor, as well as spreadsheet and database software. Official support for *Works* ended in 2012 and the software isn't available for purchase through official channels.

Microsoft did make some half-hearted attempts to help users who'd switched to the official *Office* suite, including limited support in *Microsoft Word* and *Excel* for opening newer types of *Works* files.



If Works word processor documents don't open correctly in Writer, use wps2odt to configure encoding for the output document.

Still, this isn't much help these days to users who have older files in this proprietary format. Fortunately, the open source community has proved equal to the task through the development of the libwps C++ library, incorporated into software such as *LibreOffice.*

Microsoft's *Money* suite also enjoyed a brief vogue between 1991 and 2010, allowing users a way to store information like account balances and investments. Support was officially dropped for the software and its proprietary (MNY) format but the tech giant did release a 'sunset' application to open and export the data.

In this guide, we discuss how you can use libwps as implemented in *LibreOffice* to open *Microsoft Works* documents. You'll also discover how to import account information from *Microsoft Money* into Linux apps such as *GNUCash.*

Microsoft's desktop publishing program *Publisher* has been around since 1991. It's currently available with various *Microsoft 365* (formerly *Office 365)* subscriptions, but as of October 2026, support will be discontinued. For this reason, we've covered how to export *Publisher* files in EPS format, as well as how to open them in *LibreOffice Draw (see boxout, left).*

We were able to get both *Microsoft Works Suite 2002* and *Microsoft Money 2004* running in a Windows 11 virtual machine on Linux.

This remains the best way to preserve as much of the original file formatting as possible, as both *Works*

» PROPRIETARY PUBLISHER PLANS

While it's not a legacy app just yet, you can prepare for the inevitable by exporting existing *Microsoft Publisher* projects. You can save pages as HTML/PDF, but if you want to print *Publisher* pages professionally, it's best to use EPS (Encapsulated Postscript) format.

If you still have access to *Publisher,* do this by going to File > Print. In the Print dialog box, select Print Setup > Properties. Choose EPS as the PostScript output format. For best results, save each page individually. You can configure options for this from Print To File.

If you no longer have access to a Windows machine and/or don't want to pay £59.99 for a *Microsoft 365* subscription to open your old PUB files, you can also access them via *LibreOffice Draw.*

During our tests with documents created in *Publisher 2002,* we found *LibreOffice* rendered them faithfully both when being opened and when converted to ODF Drawing (ODG) format.

and *Money* support exporting files to more universal formats such as DOC and QIF.

If this isn't feasible, your best bet for opening *Works* files is to have a recent install of *LibreOffice*. This suite comes installed in many popular distros, such as Ubuntu.

If you don't have it already, you can follow the steps on the *LibreOffice* wiki to install the correct version for your particular distro: **https://wiki.documentfoundation.org/Documentation/Install/Linux**.

Microsoft Money is very glitchy under Wine but should be sufficient to let you export your data in Quicken (QIF) format.

Word processor documents

Once *LibreOffice* is installed, technically you can right-click WPS documents to choose Open With, but on our Ubuntu 24.04 test machine, the only option offered was the *Calc* spreadsheet software.

Instead, open *LibreOffice Writer*. From here you can then open the WPS file. Once this is done, choose File > Save As, then select a more flexible format, such as Open Document Text (ODT).

During our tests, we found the WPS document opened readily enough – but the previously colourful formatting was gone.

If you experience similar issues, you can also convert WPS files via the command-line utility *wps2odt*. To get started, first install via the terminal:

```
$ sudo apt install wps2odt
```

Once this is complete, use *cd* to go to the folder where your WPS document is located. You can now convert the file to a different format by specifying both the input and output file – for example:

```
$ wps2odt Editor.wps Editor.odt
```

If the output document doesn't display correctly, you can run:

```
$ wps2odt –list-encoding
```

This lists various types of encoding, which you can attempt to use for your output document, such as:

```
$ wps2odt Editor.wps --encoding CP875 Editor1.odt
```

Other Works documents

Once you've mastered opening WPS documents in *LibreOffice Writer*, you can do the same with the corresponding applications for other proprietary *Works* formats.

For instance, *Works* spreadsheets (XLR format) open seamlessly in *LibreOffice Calc*, although you may want to change the default Comic Sans font to something a little more appealing before you use Save As to save your spreadsheet in a more universal format, such as Open Document Spreadsheet (ODS).

LibreOffice Base can also open *Works Database* (WDB) files but with the caveat that it only uses table view. In other words, the raw data is preserved but isn't displayed in jazzy interactive forms.

Money talks

Former users of *Microsoft Money* will remember that once you got past the constant demands to sign up for MSN services and on-screen narration, the software was perfectly serviceable for personal finances.

Although the proprietary MNY format is no longer supported by Microsoft, after discontinuing the software in 2010 the tech giant did release *Money Plus Sunset*. The program doesn't contain all the features of the original software, but it's free and is sufficient for exporting MNY data in a more accessible format.

To get started, you need to download *Wine* via the terminal:

```
$ sudo apt install wine64
```

Next, enable 32-bit architecture support:

```
$ sudo dpkg --add-architecture i386
```

You can now add the *Wine* repo suitable for your distro – for example:

```
$ sudo add-apt-repository 'deb https://dl.winehq.org/wine-builds/ubuntu/ noble main'
$ wget -O - https://dl.winehq.org/wine-builds/winehq.key | sudo apt-key add -
$ sudo apt update
$ sudo apt install --install-recommends winehq-devel
```

Microsoft no longer offers the *Sunset* app for download but we were able to get our hands on it via the Internet Archive:

```
$ wget https://archive.org/download/MSMoneySunset/USMoneyDlxSunset.exe
```

Once the download is complete, right-click and select Open With Wine Windows Program Loader. The software installs the necessary dependencies. You can now use the terminal to launch the *Sunset* program:

```
$ cd ~/.wine/drive_c/'Program Files (x86)'/Microsoft\ Money\ Plus/
$ wine msmoney.exe
```

On first launch, click Next to open an existing file. *Wine* assigns drive letter Z to the root drive – so, for instance, we were able to access the *Microsoft Money* project file in our documents via:

```
Z:\home\nate\Documents\My Money.mny
```

Once the project file is open, go to File > Export. Next, select Strict QIF. This ensures the exported data is accessible to other applications that support the *Quicken* format.

You can choose to save info for all your accounts, or just your investments. Make your choice using the interactive forms. Next, give the file a meaningful name, such as **budget.qif**.

The resulting file can be opened in various native Linux finance apps. We recommend *GNUCash*. LXF

QUICK TIP

To import *Quicken* files in ***GNUCash***, simply install the app via Software. Next launch and choose File > Import > Import QIF. Once you've selected the file, choose Forward to configure import options.

LIBREOFFICE DRAW

Credit: The Document Foundation

Advanced PDF edits

Nate Drake delves into the very best Linux apps for creating, managing, editing and signing the previously proprietary PDF format.

OUR EXPERT

Nate Drake encountered problematic PDFs for the first time when a DRM-protected copy of *The Time Machine* wouldn't open in Ubuntu.

Since 1998, PDF (Portable Document Format) files have been available as an open standard, no longer subject to the proprietary whims of Adobe. Still, even after a quarter of a century, *Adobe Reader* remains one of the most popular programs for viewing and annotating PDF documents. Although there are open source Linux alternatives, not all of these have the comprehensive features offered by Adobe software.

In this guide, we're going to explore some popular options for viewing, creating and editing PDFs in Linux. We'll also touch on more advanced features, such as digital signatures.

Taking the easy way out

Although there's no native *Adobe Reader/Acrobat* software for Linux, the company does offer an online PDF editor. To get started, point your browser to **www.adobe.com/acrobat/online/pdf-editor.html**.

This is certainly one of the simplest tools for managing PDFs, in that the editor supports adding text and sticky notes, highlighting text and freehand drawing. Drawbacks include the fact that you have to upload potentially sensitive documents to Adobe's website. You also need to register an Adobe account using your email or a Facebook, Google or Apple ID.

If you're not concerned about using proprietary tools and have the cash to spare, Adobe's online PDF editor is likely to have everything you need. You can also click the Download button on PDFs to save a revised version of the document.

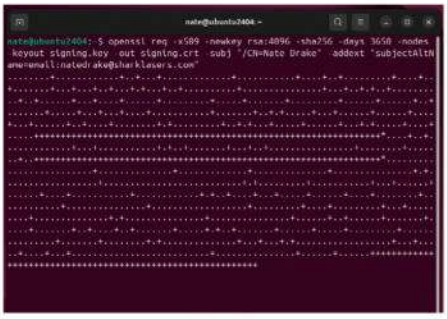
Filing with Firefox

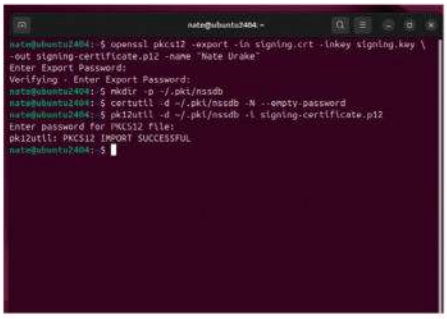
If you prefer a free, open source alternative, *Mozilla Firefox* has an excellent integrated PDF editor. Just click the link to a PDF via the browser. If the PDF is saved to your system, right-click to Open With *Firefox*.

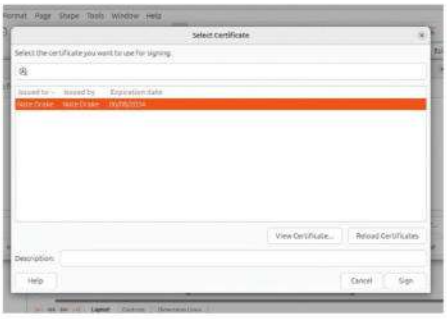
The options along the top toolbar are easy to understand. Click into the icon at the top-left to show the sidebar. By default, this displays all document pages but the icons at the top also enable you to apply additional filters to display attachments and layers.

The icons along the top-right have options to highlight text in multiple colours, add text boxes and

DIGITALLY SIGN PDFS IN LIBREOFFICE







1 Generate your certificate

Launch the terminal and install the prerequisites with `sudo apt-get update && sudo apt-get install openssl libnss3-tools` .

Generate your X509 certificate via OpenSSL, for example:

```
openssl req -x509 -newkey rsa:4096
-sha256 -days 3650 -nodes \
-keyout signing.key -out signing.crt -subj
"/CN=Nate Drake" \
-addext
"subjectAltName=email:natedrake@
sharklasers.com"
```

2 Convert your certificate

Convert your certificate to p12 format, for example:

```
openssl pkcs12 -export -in signing.crt
-inkey signing.key \
-out signing-certificate.p12 -name "Nate
Drake"
```

Next, create your NSS database:

```
mkdir -p ~/.pki/nssdb
certutil -d ~/.pki/nssdb -N --empty-
password
pk12util -d ~/.pki/nssdb -i signing-
certificate.p12
```

3 Sign your PDF

Open *LibreOffice*. Choose Tools > Options > Security. Click Certificate > Select NSS Path and make sure that it's set to **~/.pki/nssdb**. Choose Apply, then go to File > Digital Signatures > Sign Existing PDF. Open your chosen file, then select Sign Document. Click Sign Document once again, select your certificate and then click Sign.

draw freehand, as well as add and edit images. From here you can also print and save the edited document to your machine.

Opening with Okular

If you do use *Firefox* to save a PDF locally in Ubuntu, you'll notice by default it opens in Gnome's *Evince* document viewer.

This is an excellent way to view PDFs but it has no editing features. Naturally, you can just keep using *Firefox* but KDE's *Okular* offers a number of superior PDF features. These include annotation and highlighting, as well as the ability to copy text and images.

If you're using Ubuntu, *Okular* is available for installation via *Gnome Software*. The program is also available via Flathub (**https://flathub.org/apps/org.kde.okular**).

On first launch, *Okular* is set to Browse mode, enabling you to scroll through the pages of your PDF.

To copy text, simply scroll to a suitable paragraph, then hold Ctrl+4 to activate text selection. Use your mouse to click and drag over the required text, then right-click and select Copy Text. You can also convert the entire PDF document to a text document by selecting File > Export As > Plain Text.

If you want to copy an entire section – for example, one containing images – hold Ctrl+3 to switch to area selection. Click and drag around the area to open a context menu. From here, you can either select Copy To Clipboard or Save To File.

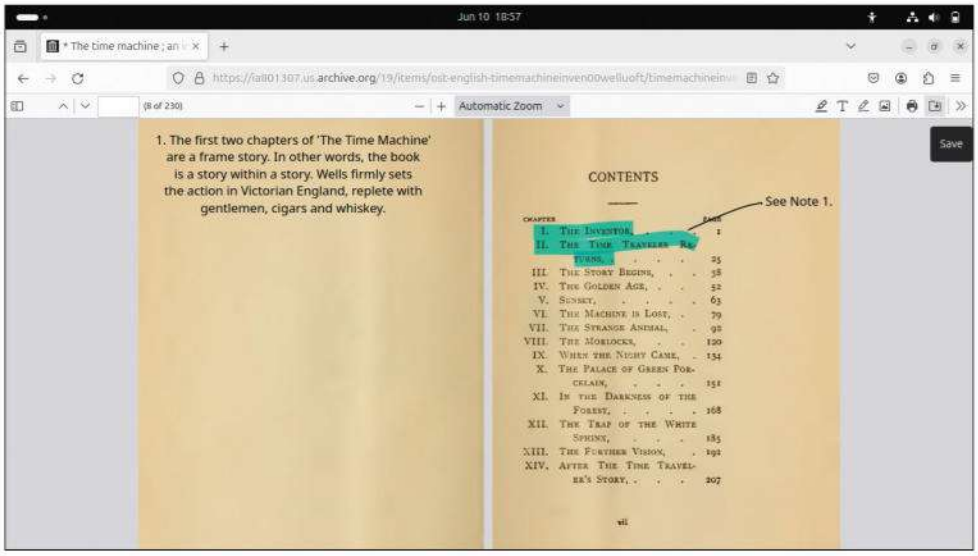
Right-click the drop-down arrow at the very top-right of the *Okular* window to select Show More Annotation Tools. From here you can edit text via underlining/strikeout, as well as add pop-up or inline notes. The Quick Annotation drop-down menu at the top-right contains commonly used tools. These can be selected via numerical hotkeys – for example, pressing `1` switches to the yellow highlighter.

Digitising with Draw

LibreOffice Draw is primarily designed for graphics editing but also manages PDFs well. Some distros, such as Ubuntu, come with *LibreOffice* installed. The suite is also available in the repositories of many others. You can also follow the instructions at **https://wiki.documentfoundation.org/Documentation/Install/Linux** to install *LibreOffice* manually.

When you first open a PDF, you'll notice that you can simply click on a section of text to edit it with ease. However, we noted that *Draw* doesn't play nicely with previous modifications made with other programs, such as *Okular*.

Draw can sometimes also place text boxes and other objects in a different area from the original file. You can click and drag them back into place via the mouse. You can also select a group, then choose Align Objects along the top toolbar to put them back in order. From here you can also enable Helplines While Moving to display gridlines as you move objects.



Firefox has excellent PDF markup tools, including highlighting. You can also add images and save the modified PDF.

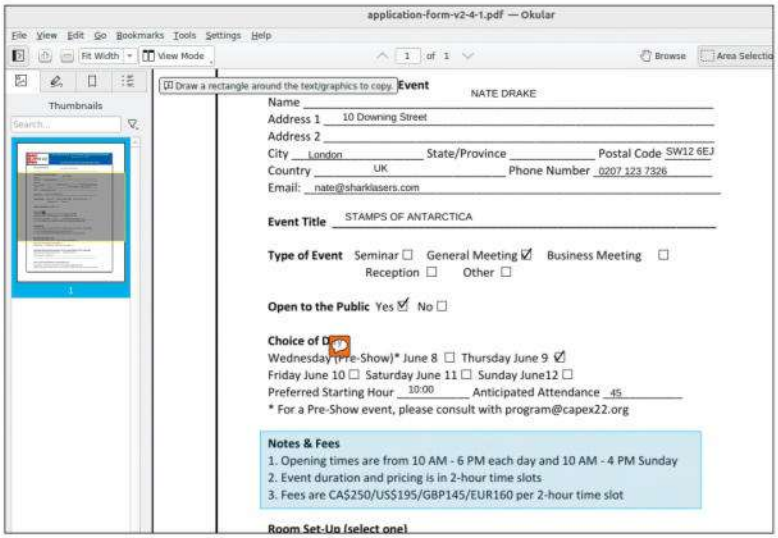
Given *Draw's* creative approach to formatting, we suggest making any text edits first before annotating/marking up the document. Once this is done, go to File > Export As > Export As PDF to save the modified file.

The left-hand toolbar contains other editing options, such as inserting shapes and symbols. If you open the context menu under Curves And Polygons, you can also choose Freeform Line to write a basic signature into the PDF.

If you want to use a digital signature without signing up for Adobe, technically you can create your own NSS (Network Security Services) certificate via the CLI, then import it into *Okular* or *Draw*. X509 certificates are only useful if they can be independently verified by a certificate authority like Verisign. However, you can follow the steps in the walkthrough to generate your own certificate via OpenSSL if you wish. LXF

QUICK TIP

If you're viewing a file or web page in *Firefox* and want to save it as a PDF, simply hold Ctrl+P to summon the print dialog. In the Destination drop-down menu, choose Save To PDF, then hit Save.



Okular has handy tools for copying areas, so you can paste text and images into another program. You can also add notes and highlights.

Secure file transfers – not sexy but important

The perma admin-sexy **Stuart Burns** takes you through the fundamentals of securely transferring important files around systems.

Transferring files is not the most interesting subject in Linux administration but it is important. Sometimes as an administrator, you need to transfer large amounts of files to a remote system. Where one method fails, another may work.

There are several ways to duplicate data between systems efficiently. Below we demonstrate some of the methods and the scenarios where each has a best fit.

Clue's in the name

SCP is the great grandfather of securely (encrypted) copying data between two systems. It is OK for small, quick file copies but it is showing its age. It is inefficient and not very communicative regarding progress.

SCP (Secure Copy Protocol) is the default method for copying files between hosts without installing additional tools. It's built in to most Unix/Linux-like systems. It relies on the SSH server working on both source and destination. The underlying OpenSSH server takes care of the encryption. To copy a file from the local system to the remote system is as simple as:

```
$ scp myfile.txt sysadmin@remotehost:/home/sysadmin/
```

To copy multiple files, use the following:

```
$ scp *.txt sysadmin@remotehost:~
```

Every time this is run, it recopies and overwrites existing data. There must be an easier way. Enter *rsync*.

The daddy of copy

Rsync is the ideal tool for copying substantial amounts of data. It has many advantages over SCP but by far the major differentiator is that *rsync* can resume a halted or failed copy operation. That makes it ideal for unreliable connections with lots of data copying.

```
presets/Contrastive Search.yaml
          45 100%    0.22kB/s    0:00:00 (xfr#409, to-chk=18/501)
presets/Debug-deterministic.yaml
          26 100%    0.13kB/s    0:00:00 (xfr#410, to-chk=17/501)
presets/Divine Intellect.yaml
          65 100%    0.32kB/s    0:00:00 (xfr#411, to-chk=16/501)
presets/LLaMA-Precise.yaml
          63 100%    0.31kB/s    0:00:00 (xfr#412, to-chk=15/501)
presets/Midnight Enigma.yaml
          66 100%    0.32kB/s    0:00:00 (xfr#413, to-chk=14/501)
presets/Null preset.yaml
          15 100%    0.07kB/s    0:00:00 (xfr#414, to-chk=13/501)
presets/Shortwave.yaml
          65 100%    0.32kB/s    0:00:00 (xfr#415, to-chk=12/501)
presets/Yara.yaml
          65 100%    0.31kB/s    0:00:00 (xfr#416, to-chk=11/501)
presets/min_p.yaml
          12 100%    0.06kB/s    0:00:00 (xfr#417, to-chk=10/501)
presets/simple-1.yaml
          63 100%    0.30kB/s    0:00:00 (xfr#418, to-chk=9/501)
prompts/
prompts/Alpaca-with-Input.txt
```

Rsync can copy files locally, as shown. Useful for big copy operations.

It is available on pretty much all Linux systems but to install it if it's not, just use your packaging tool. For example, on a Ubuntu-based tool, it would be:

```
$ sudo apt install -y rsync
```

Additionally, *rsync* can compress files during the copy process. This can be faster than a standard copy because it can compress data on the fly. *Rsync* is intelligent enough to copy only the changes between the source and destination. It doesn't blindly copy the data but does a mathematical comparison on both sides to ensure integrity and accuracy.

That means that if *rsync* is used as a form of backup, it doesn't recopy everything when the backup is run, just the changes since the last copy, referred to as deltas. That makes it ideal for saving copies of your **home** directory without resending the entire folder contents every night.

A simple example of using *rsync* could be:

```
$ rsync -azvp /home/* sysadmin@remoteserver:/backup --progress
```

By default, *rsync* output is very minimal, so copying multiple gigabytes to a remote server can make it look as though it has stalled. The **-p** option gives far more detailed output and tells you what file is being copied. However (and in our humble opinion), the **--progress** flag should be used instead. It gives a much more detailed breakdown of the file copy as it happens. The other switches are:

a – Archive everything, essentially meaning copy all subdirectories and so on

REGRETTING REGRESSHION

OpenSSH, regarded as one of the world's "most secure software implementations", has a "glaring gap" that allows threat actors to completely take over Linux systems that have it installed, experts have warned.

A report from Qualys claims the vulnerability has been present for four years, and affects 14 million end points worldwide. Qualys dubbed its finding regreSSHion, and says it is tracked as CVE-2024-6387. The flaw was so named because it's a regression of the previously patched vulnerability CVE-2006-5051, fixed in 2006. A regression is a flaw that was fixed but later reintroduced. Get the report here and patch your servers: **https://bit.ly/lxf318ssh**

z – Enable compression
v – Verbose output (which is kind of minimal even at best)

If using *rsync* for large copies on the terminal, which can tie up a terminal window for a long time, use *screen* or *tmux* commands to push the *rsync* copy to a virtual terminal that will persist after the virtual terminal is logged off.

This one *rsync* command can do most of the heavy lifting required for most scenarios. At the same time, *rsync* isn't limited to just copying to remote servers; it can be used locally to make copies of folders. A recent example was when we had to move a large application to a different partition. There are many ways to do the copy, including the old *cp* command, but using *rsync* meant the copy of the hundred-gig-plus of data was repeatable and also preserved the ownership of the files after the copy had been completed.

```
sysadmin@lf318: ~
presets/Null preset.yaml
          15 100%    0.03kB/s    0:00:00 (xfr#414, to-chk=13/501)
presets/Shortwave.yaml
          65 100%    0.13kB/s    0:00:00 (xfr#415, to-chk=12/501)
presets/Yara.yaml
          65 100%    0.13kB/s    0:00:00 (xfr#416, to-chk=11/501)
presets/min_p.yaml
          12 100%    0.02kB/s    0:00:00 (xfr#417, to-chk=10/501)
presets/simple-1.yaml
          63 100%    0.12kB/s    0:00:00 (xfr#418, to-chk=9/501)
prompts/
prompts/Alpaca-with-Input.txt
         222 100%    0.44kB/s    0:00:00 (xfr#419, to-chk=8/501)
prompts/QA.txt
          63 100%    0.12kB/s    0:00:00 (xfr#420, to-chk=7/501)
training/
training/datasets/
training/datasets/put-trainer-datasets-here.txt
```

Rsync copying files but without --progress. Note the lack of useful information.

Had the same been done with *cp*, every subsequent copy would start from scratch and the volume of data would be 'everything, every time', rather than just what had changed. Using it locally, the command would be:

```
$ rsync -a dir1/ dir2 --progress
```

Whilst *rsync* is great, it does have one or two caveats. Firstly, it requires ports 873/TCP and 22/TCP to be open for it to run. However, with it being Linux, there is always a way around the issue. This is to use the following switch, which uses secure shell port 22:

```
$ rsync -rvz -e 'ssh -p $port' ./dir user@host:/path
```

Putting it all together, *rsync* and key-based logins can be used to make life much easier. Key-based setup is an entire topic in itself but we'll quickly demonstrate how useful it can be. On the source-based system, use the following command, and when asked for a password and password confirmation, press Enter:

```
$ ssh-keygen
```

This creates a public key (assuming you haven't already generated one). The public key pair keygen process creates a public key and a private key. Setting it up on the remote system uses the *ssh-copyid* command to set it up on the destination side (way easier than manually attempting it):

```
$ ssh-copyid user@<remote system>
```

It asks for the password and sets up the public portion of the key. Once done, it means the user can remotely log in using that key (so keep it very safe).

Once that is set up, it becomes straightforward to push and pull data with *scp* and *rsync* to and from remote systems. Using the public key method of copying means it becomes trivial to add the *rsync* command for backing up the **home** folder (obviously substituting values) to the user's crontab.

As we have run out of space, check in next month to see how crontab can be used effectively to automate the execution of jobs on a local Linux box. LXF

Good old scp works everywhere but each copy is a new copy, overwriting existing copies at the destination.

```
typing.py                                     100%  118KB 197.4MB/s   00:00
configparser.py                               100%   54KB 136.7MB/s   00:00
_osx_support.py                               100%   22KB  92.9MB/s   00:00
copyreg.py                                    100% 7677    48.2MB/s   00:00
traceback.py                                  100%   40KB 147.4MB/s   00:00
this.py                                       100% 1003     9.3MB/s   00:00
sunau.py                                      100%   18KB  81.4MB/s   00:00
__init__.py                                   100%   97   941.6KB/s   00:00
__init__.cpython-311.pyc                      100%  683     6.5MB/s   00:00
spam.cpython-311.pyc                          100%  679     6.4MB/s   00:00
spam.py                                       100%   97   957.7KB/s   00:00
libpython3.so                                 100%   15KB  73.3MB/s   00:00
```

Stuart Burns is a Linux administrator for a Fortune 500 company specialising in Linux.

» SWAPPING OLD HATS

Canonical has released a long-term-support Docker image to rival the Red Hat Universal Base Image (UBI). The Canonical press release was quite short on details but there are differences.

The Red Hat UBI is incredibly popular in enterprise circles because it provides a free as in beer highly consistent long-term container image that can run on any Dockerised platform and includes all the Red Hat design admins have grown used to. A lot of DevOps platforms use it, as it's about as stable as you can get – just add your own batteries (software) to the Docker file.

Canonical is taking the fight to Red Hat because up until now it was the missing piece of the puzzle. Enterprises move very slowly and sometimes even LTS time windows aren't enough.

A lot of enterprises and SMEs are looking to move because either they don't like the Red Hat business model or, more likely, they just can't afford it.

Canonical is promising a minimum of 12 years of support, 24-hour fix windows for major vulnerabilities and the promise it will continue to work on all major hypervisors and cloud platforms.

The fly in the ointment appears to be that this image isn't free to all. It's for companies that want long-term stability, and are prepared to pay for it and for Canonical to handle a lot of the lifting.

The new Canonical-devised base image adds a missing piece of the puzzle and the oddity of adding a Red Hat container on a Ubuntu Docker host.

If it is 'pay to play' on Canonical's part, it represents a huge missed opportunity to take the fight to Red Hat.

OVHcloud

No one can say **Mike Williams** doesn't have his head in the clouds as he tests a cloud-hosting provider geared towards tech-savvy users.

IN BRIEF

OVHcloud supplies its customers with a variety of competitively priced cloud-based necessities, including VPS, dedicated servers, bare-metal servers, hosted private cloud, public cloud, as well as hybrid solutions. It isn't as geared towards beginners as we would have liked, its customer support needs a revamp, and the lack of a refund period means you can't try out the service risk-free.

Founded in 1999 OVHcloud is a French web-hosting company with an international presence thanks to over 400,000 servers spread across more than 43 data centres in four continents. As the largest provider in Europe today, OVHcloud offers a wide range of hosting products.

OVHcloud is one of the cheapest, if not the cheapest, providers when it comes to shared web hosting. Plans start at just $1.04 per month for the Starter Hosting tier, which gets you 1GB of disk space, a free domain name for the first year, support for one website, free SSL certificate, anti-DDoS protection, and one-click *WordPress* installation.

Interestingly, the shared hosting plans do not come with CDN included; perhaps that's why they're so cheap. CDN, which reduces the loading time for your pages, costs an extra $2.79 per month.

With OVHcloud, you can spend anywhere between $50 and $4,000 on a dedicated server. The cheapest comes with the Intel Xeon E Processor, 32GB RAM, and 1TB SSD NVMe storage. This is an excellent package and right up there with the very best.

OVHcloud offers a ton of VPS server options, so choosing one is admittedly not as simple as, say, DreamHost or Liquid Web. Where OVHcloud really sets itself apart is that all of its VPS servers are part of its cloud environment, so you benefit from auto-scaling, which enables you to only pay for what you use.

OVHcloud is a highly versatile and customisable hosting provider. Servers with DDoS protection and quick scaling for hosting large-scale video games? Check. Servers that facilitate remote learning and collaboration for the education industry? Check. *Minecraft* VPS servers – the kind you get with industry leaders such as Hostinger? Check.

Similarly, it has a compliant and secure solutions for just about every industry, whether that's healthcare, IT, non-profit, or the government sector. It also carries the proper attestations and certifications (HIPAA, HITECH, PCI DSS and so on) to be able to be this diverse.

Unlike other hosts, OVHcloud doesn't provide any free trials or money-back guarantees, which means that if you decide to cancel, you can say goodbye to any money you've invested so far, including setup fees.

When setting up your account, you get access to OVHcloud's dashboard, from which you can complete your purchase if you haven't done so already. From here, you should be able to monitor and manage all aspects of your account, such as passwords, backups, databases and so forth. It's worth noting that we didn't use the word "should" because of a lack of belief in your abilities but due to the somewhat slow, painfully plain and not too novice-friendly OVHcloud dashboard.

An abundance of data centres scattered across the globe typically predicts lightning-fast speeds, and this is true with OVHcloud. A GTmetrix test performed on OVHcloud's main site showed that it took 2.9 seconds for it to fully load, whereas the recommended time is less than 3.5 seconds (and the average is 8.1). All other web vitals showed a presentable performance, building up to a near-perfect A (95%) as a final GTmetrix grade.

Of course, our favourite penguin is fully supported here.

OVHcloud offers a 99.9% uptime guarantee backed by SLA across most of its VPS ranges. Rivals such as InMotion Hosting offer a 99.999% uptime guarantee, whereas Liquid Web and DreamHost lead the space with a whopping 100% uptime guarantee. During a two-week-long uptime test on OVHcloud's main site (via UptimeRobot), we only recorded a single case of downtime, but that lasted for 15 minutes straight.

Support could do with work – the knowledgebase contains plenty of step-by-step guides, but there's barely anything to help beginners. On the other hand, the community forum seems quite active.

Standard is the only level of support that comes with every hosting solution without additional charges. You are free to contact the sales team via telephone (weekdays 9am-5pm) and email/ticket. However, don't expect to get anything but basic information here. LXF

VERDICT

DEVELOPER: OVHcloud
WEB: www.ovhcloud.com
PRICE: Various

FEATURES	7/10	EASE OF USE	7/10
PERFORMANCE	8/10	VALUE	8/10

Although OVHcloud's cloud hosting solutions have managed to put an affordable price tag on its top-notch services, a lot of standard features are lacking.

» Rating 7/10

CREDIT: OVHcloud

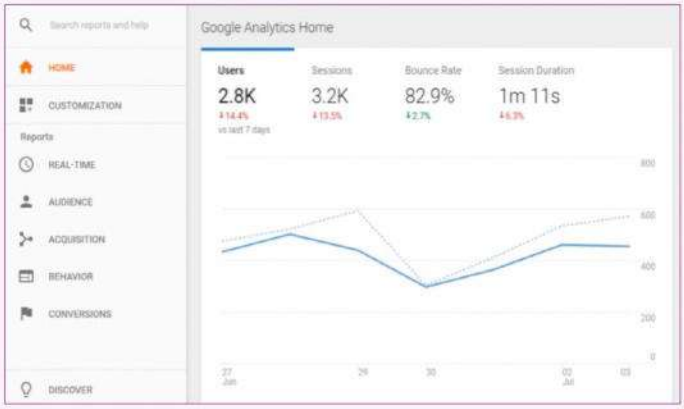
Google SEO Tools

Free SEO tools from the king of search engines has **Michael Graw** wowed with their power.

IN BRIEF

The trifecta of Google Analytics, Search Console and Ads is an extremely powerful combination for website owners. The three tools together allow you not only to monitor your website traffic, but also to build more traffic through organic and paid search results. The only thing lacking is information about how your website is ranking in search results for specific keywords.

Google is the king of search engines, and there is no doubt that most of the SEO industry revolves around it for marketing and sales. For this purpose, Google has a set of tools that help digital marketers analyse their websites, study their performance and improve their visibility on the SERP. Unsurprisingly, Google delivers a stellar performance in this segment, too. With a range of dynamic and useful tools, it helps its users do a lot with their websites. And some of its tools, like Google Analytics, even offer many insights for free.



Google Analytics is a web analytics service that tracks and reports website traffic.

Google Analytics tracks and reports website traffic. It's one of the most used analytics services on the web. It provides detailed insights about a website's visitors, including how they interact with the pages of the site. Plus, it tracks everything from page views to bounce rates and conversion paths, giving you clear insights.

Recently, Google rolled out Google Analytics 4 (GA4), the successor to Universal Analytics (UA). This comes with a new approach to how data is collected, processed and reported compared to its predecessor. It is so detailed that you not only see how many people visit your site but how they found you, which pages they're interested in, and how long they stick around.

Plus, there's cross-platform tracking that helps you see the full picture, showing how your audience moves between devices, such as phones, tablets or laptops.

Google Analytics also integrates seamlessly with Google's advertising platforms. This is great for targeting your ads more effectively. You can see which ads drive traffic and conversions, helping you focus your efforts and budget on what really works. GA4 is also designed with privacy in mind, ensuring compliance with current regulations.

Google PageSpeed Insights helps evaluate the performance of specific pages of a website. It measures the speed and usability of a site on mobile and desktop devices, and delivers a score between 0 and 100. This can help identify issues that slow down your site and offers a detailed breakdown of factors such as loading times, interactivity and the stability of content as it loads.

Google Keyword Planner is designed for advertisers to discover keywords related to their businesses and see estimates of the searches they receive and the cost to target them. Its direct integration with Google Ads provides a streamlined workflow for creating and refining ad campaigns based on robust keyword data.

One of Google Keyword Planner's best features is its ability to offer comprehensive data on keyword search volume, competition level and cost-per-click (CPC) estimates. This data is vital for advertisers and marketers about which keywords to target, how to allocate budgets effectively, and what kind of content or ads might resonate with their target audience.

Today, the majority of companies that want an online presence are available on Google My Business (GMB). GMB helps business owners manage their digital footprint across Google Search and Maps, sharing information such as location, contact details, business hours and services.

Perhaps the biggest issue with Google's SEO interface is that Analytics, Search Console and Ads are three different platforms. You can link Search Console and Ads, but you still need to navigate back and forth between the two interfaces for most tasks.

Google offers support for Analytics and Search Console by web only. Both platforms have extensive documentation centres, and you just need to describe your issue to find the appropriate help file. If you get stuck, though, support is limited to posting in a help forum and hoping another user answers your question.

Support for Ads is more concrete. There's an online documentation library but you can also get help over the phone, by live chat or by email. LXF

VERDICT

DEVELOPER: Google
WEB: https://search.google.com/search-console
PRICE: Free

FEATURES	**9/10**	**EASE OF USE**	**9/10**
PERFORMANCE	**10/10**	**VALUE**	**10/10**

Google Analytics, Search Console and Ads are free to use and extremely powerful. Every website owner should be incorporating Google SEO tools into their web strategy.

» **Rating 10/10**

CREDIT: Google

HotPicks

Parsec » Mpv » Morphosis » Warp » Fotema » Ventoy » StackAndConquer » iQPuzzle » Qutebrowser » RSS Guard

Mayank Sharma
burned to a crisp when he went to vote, and stayed up past bedtime to watch cricket matches in EST, but still dug up software gems.

CLOUD FILE SHARING

Parsec

Version: 2.17.0
Web: https://parsec.cloud/en/

Parsec is a Dropbox-like online file-sharing service, which prioritises security, and is geared towards collaboration and file transfer. Its Linux client is distributed as a Snap package and is installed with `sudo snap install parsec --classic`. On first launch, the app prompts you to add a device, which involves creating an organisation. Besides the identifying information, you're asked if you want to collaborate through Parsec's servers. You do have the option to roll out your own, but that takes some doing.

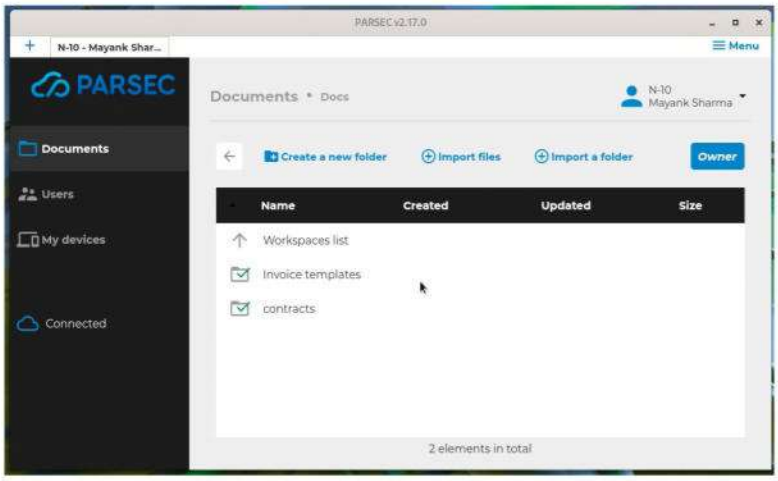
You're then asked to choose an authentication method. The no-cost version only offers password authentication. Parsec encrypts data with this password, so don't forget it. It does allow you to create a recovery device that helps you regain access to your data in case you misplace the device

You're now taken to the app's main interface. In Parsec, data is stored in workspaces, and each has its own policy for read and write access. Head to Create A New Workspace to get started. The workspace creator automatically becomes its owner, and can invite others to join with one of the four self-explanatory permissions: reader, contributor, manager and owner.

Once you've created a workspace, it appears in your distro's file manager as a mounted folder. To unmount and remount the folder, you have to head to the Parsec client, which displays the toggle at the bottom-left of the workspace card.

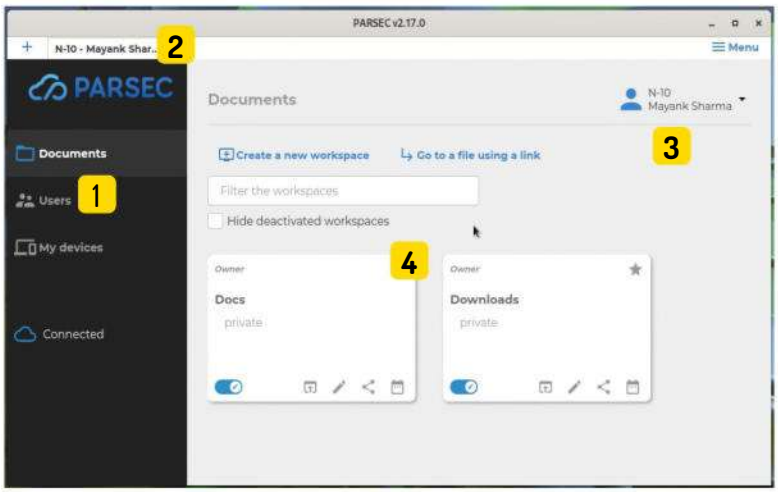
Besides your distro's file manager you can also use Parsec's web-based file manager. To bring it up, click the mounted workspace. Now copy some data into the workspace, then head to the Users tab to invite users to this workspace. Once added, all users can share the files and folders housed inside the shared workspace. Depending on permissions, they can add data inside this folder, which is then propagated to all other users.

The Parsec client can also work while offline, and synchronises data automatically as soon as it can communicate with the server.



Parsec offers a free plan with 10GB of storage that you can share between three users. Paid plans offer more storage space and a lot more features.

LET'S EXPLORE PARSEC...



1 **Navigation bar**
Switch between the major Parsec functions: **D**ocuments lists the workspaces; Users lists the number of connected users; and with My **D**evices you can browse existing devices and add new ones.

2 **Tabbed interface**
You can open a new tab to create a new organisation, join an existing one, or recover a device using the recovery disk.

3 **User details**
Click here if you need to change the authentication of the user, update to a paid plan, and view all the kinds of information about the current organisation.

4 **Workspace cards**
This is where all the shared workspaces are listed. You get controls to mount and unmount a workspace, along with buttons to share and rename a workspace.

MUSIC PLAYER

Musicpod

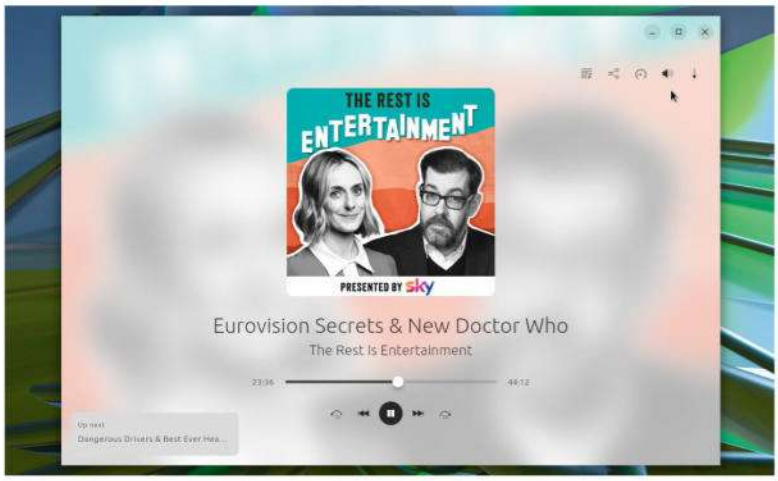
Version: 1.4.4 **Web:** https://github.com/ubuntu-flutter-community/musicpod

Although most Linux distros ship with an app to play music, the Linux ecosystem has tons of apps for playing audio. *Musicpod* distinguishes itself as a one-stop shop for playing local audio, online radio and podcasts.

The app is available on Flathub and can be installed with `flatpak install flathub org.feichtmeier.Musicpod`. The Flatpak version of the app doesn't work well on Fedora, so the developers suggest grabbing the app with Snap. You first have to install Snap support in Fedora with `sudo dnf install snapd`. Then either log out and back in again or restart your system to ensure Snap's paths are updated correctly. Once you're back in, install *Musicpod* with `sudo snap install musicpod`.

When you fire up *Musicpod*, the app automatically pulls in all the music from under **~/Music**, and lists it alphabetically. Depending on the size of your music library, it could take a while to import all your tracks.

If you have correctly tagged your music, in addition to the alphabetical list, you can also view your music by Artist, Albums or Genres. You also get an option to search through your collection if you're looking for a particular track.



Musicpod's audio player is the standard fare, with all the usual playback controls. It can run full-screen as well, which surprisingly shortens the seek bar.

In the radio section, you can search for stations by their name, tag, country, state and language. You can use these options to search for all pop or oldies stations, for example, or all kinds of stations in the UK, US or any other country. It's a shame you can't combine these categories to search for all pop stations in the UK, for instance.

However, you can combine the options in the podcast section. Here you can find podcasts from a country and then break them down by category, such as News, Comedy, Education and so on.

MEDIA PLAYER

Mpv

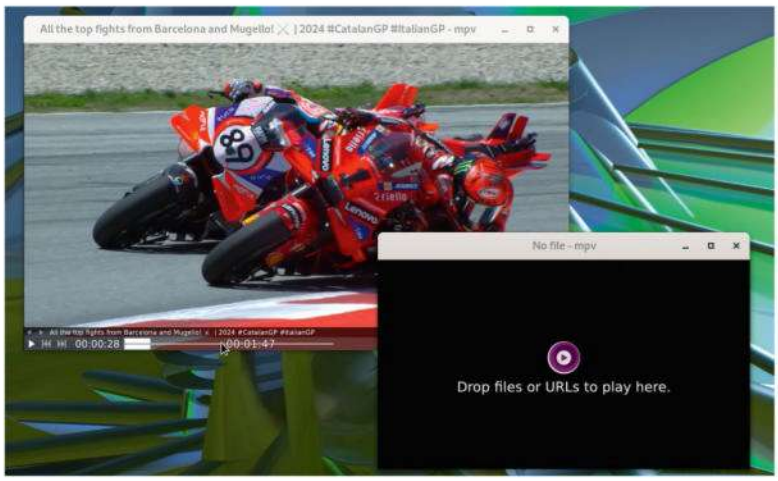
Version: 0.38.0
Web: https://mpv.io

VLC is the catch-all media player on Linux. It can do a lot of things, but you wouldn't call it lightweight. If you want to use a minimalist media player, and aren't averse to the CLI, you'll want to use *Mpv*.

The app is based on venerable CLI tools and libraries. Thanks to this, it supports a wide range of audio and video codecs, and file formats. Besides local files, *Mpv* can also stream audio and video media directly from the internet.

It is available in the repos of most distros, but it usually isn't the latest release. To get the latest stable release, the developers suggest using one of several third-party repos listed on its website. A distro-agnostic way is to grab the player through its unofficial Flathub repo with `flatpak install flathub io.mpv.Mpv`.

Once installed, you can point it to a video right from the CLI with something like `flatpak run io.mpv.Mpv ~/Videos/some-video.mp4`. If you had installed *Mpv* through your distro's repository, this command would shorten to `mpv ~/Videos/some-video.mp4`.



While Mpv is a CLI-based player, it does ship with a GUI interface of sorts. You can bring it up with `flatpak run io.mpv.Mpv --player-operation-mode=pseudo-gui`.

The video launches at its default resolution. Use the arrow keys to jump five seconds, `f` to go full-screen, Spacebar to pause, and `q` to quit. If you mouse over the video, it reveals a minimal GUI with the usual buttons to control playback.

Similarly, you can point *Mpv* to a URL instead of a local file. For instance, `flatpak run io.mpv.Mpv https://vimeo.com/channels/staffpicks/818944000` streams the relevant video.

One of *Mpv's* strongest suits is its scriptability. These user scripts are like plugins that add new features to *Mpv*, and are listed in *Mpv's* wiki. In the Flatpak version, these need to be placed under **~/.var/app/io.mpv.Mpv/config/mpv/scripts**.

DOCUMENT CONVERTER

Morphosis

Version: 1.2 **Web:** https://gitlab.gnome.org/Monster/morphosis

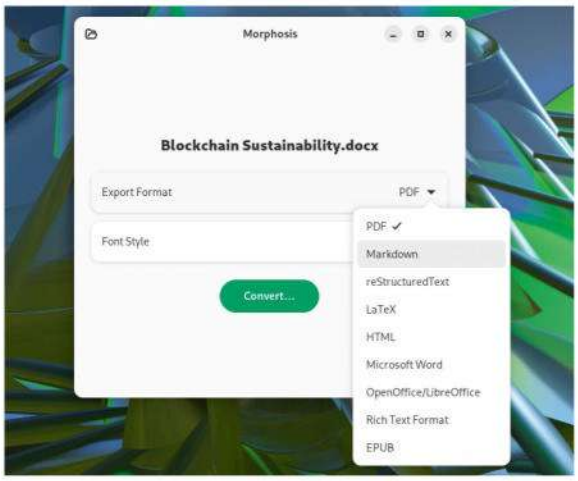
Many apps give you the option to export documents into another format. But then again, several don't. *Morphosis* is a one-stop shop that can convert documents between commonly used formats.

The app is available on Flathub and you can install it with `flatpak install flathub garden.jamie.Morphosis`.

Morphosis has the simplest of interfaces. You can drag and drop your document or use the Open Document button in the top-left to navigate your filesystem to find the document you want to convert.

After it has your document, use the Export Format drop-down menu to select the format you want the document in. This could be anything from PDF, EPUB and *OpenOffice/LibreOffice* to *Microsoft Word,* HTML, LaTeX and more.

Depending on the export format, the app also lets you select the font style of the exported document. The option is only available when exporting the document to PDF and HTML. When available, use the Font Style drop-down to select from Serif or Sans.



After Morphosis has converted and saved the new document, you get the option to open it from within the app itself.

After handing over your document to the app and selecting the export format and, if available, the font style, press the Convert button to let the app do its magic. When it's done, the app brings up a dialog to let you name the exported document and find a place to save it. By default, the app names the document after its original filename (with the exported format) inside the same directory as the original file.

While the app is very handy if you juggle with files in multiple formats, it does have limitations. For one there is no option to batch convert several documents in one go. Also, while it can export to all kinds of formats, it can't import documents in PDF or plain text files in TXT format.

FILE TRANSFER

Warp

Version: 0.7.0 **Web:** https://gitlab.gnome.org/World/warp

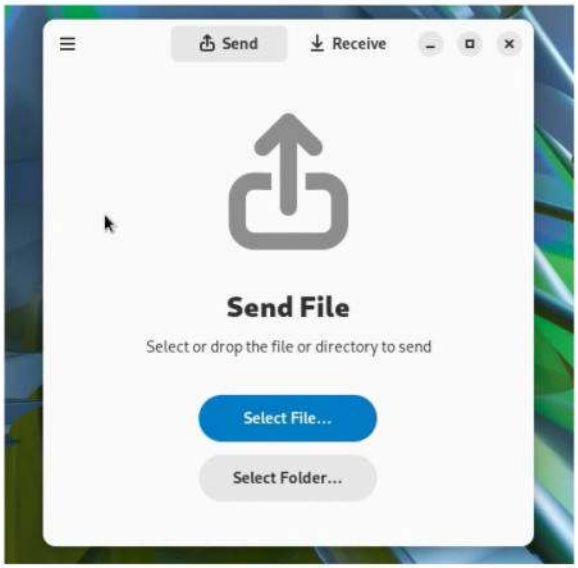
Linux has numerous tools for sending files to computers or other devices on your local network. *Warp* is another, but it can also securely ferry files to other devices and computers outside the local network via the internet.

You can install *Warp* from Flathub with `flatpak install flathub app.drey.Warp`. On first launch, it takes you through an introduction on how you can use the app to transfer files to someone else on the same network or on the other side of the planet.

The app has a straightforward interface. To send files, you don't need to go through the usual hassles that come with most cloud-based file transfer services. In fact, you don't even need to create an account.

All you need to do is ensure *Warp* is installed on both the computers that are sending and receiving files. To send a file, select the Select File or Select Folder option, and point the app to the file or folder you want to send.

This generates a QR code and a transmit code that begins with a number, followed by four words. You can then send this transmit code to the recipient. To receive



Warp is based on the Magic Wormhole CLI utility, which makes it possible to ferry files between computers irrespective of their operating systems and firewalls.

the transmission, the recipient needs to switch to the Receive section in their instance of the *Warp* app, and then enter the transmit code in the space provided to initiate the transfer.

By default, the app uses four-letter codes. But you can increase or decrease the length of the code word in the app's straightforward Preferences dialog. The other tweakable parameter in the Preferences dialog is the ability to enter a custom rendezvous server and transmit server. If you don't know what these are, it's best to let the app stick with the defaults.

IMAGE VIEWER

Fotema

Version: 1.8.0 **Web:** https://github.com/blissd/fotema

Your distro ships with an image viewer, but there are tons of other options, each with their own unique feature set. One such option is *Fotema,* which is Esperanto for "fond of taking photos". That should tell you who the app is for.

The app is available on Flathub and you can install it with `flatpak install flathub app.fotema.Fotema`.

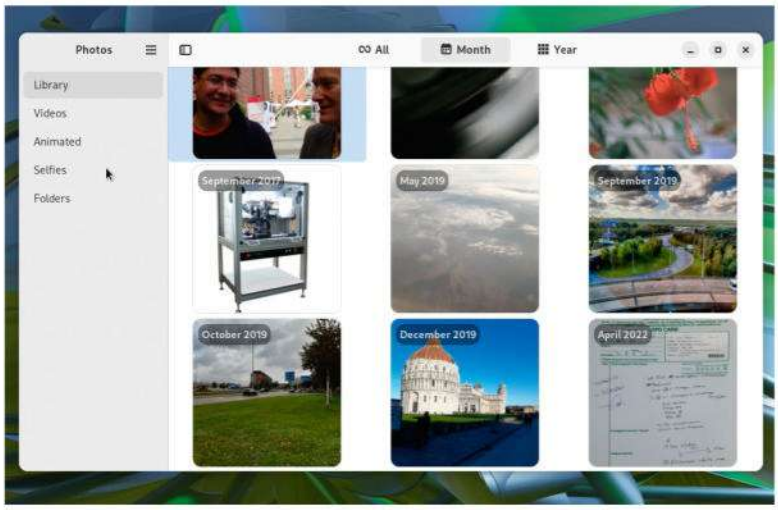
Fotema is a simple photo gallery app. It automatically reads the contents of the **~/Pictures** folder and displays its contents, burrowing down any subfolders on its own. You can sort the images by grouping them by month or year, by selecting the appropriate view at the top of the interface.

When grouped, the apps display either the month or the year, depending on the selected view. You can then click the appropriate month or year to view the individual photos.

Left-click on an image to open it. The picture has buttons to move through the gallery. There's a small 'i' button at the top-right of the interface. When toggled, the app displays some useful details about the image, read from its metadata. In addition to the filename and dates it was shot and last edited, the app also displays the original dimensions of the image, along with its format and file size.

In addition to these details, the app also offers to open the folder it resides in, using the distro's default file manager. Unless you click the 'i' button again to hide the information, the app continues displaying these details for all images.

As far as supported formats go, *Fotema* supports virtually all common image formats. Strangely, though, for an app that's apparently designed for people who love to take pics, the app doesn't support RAW images.



In addition to images, Fotema has a separate view for videos, and you can ask it to make one for selfies taken on iOS devices as well.

BOOTABLE DISK CREATOR

Ventoy

Version: 1.0.99
Web: www.ventoy.net

Running live distros off USB sticks has made Linux approachable to a whole lot of non-tech-savvy users. There's no dearth of tools that can image an ISO to a USB, but none of them is as convenient as *Ventoy.* You can just copy and paste distro ISO files into a *Ventoy*-equipped USB and you're good to go. Best of all, you can still use that USB to ferry your files.

To get started, download the *Ventoy* tarball, extract it and run `sudo ./VentoyGUI.x86_64`.

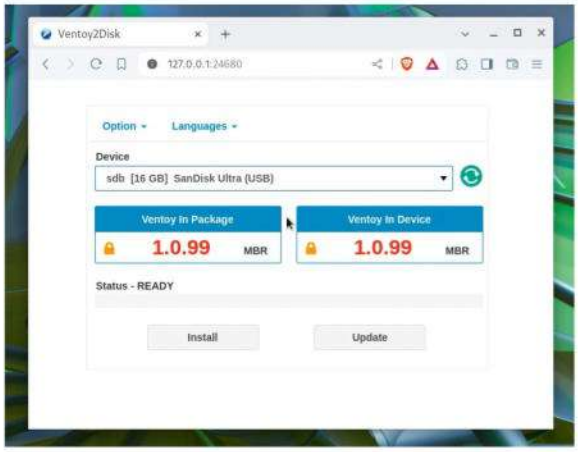
The graphical installation tool automatically detects the plugged-in USB disk. By default, it creates the devices with Secure Boot support and an MBR partitioning scheme. When you click Install, the tool formats and provisions the USB for *Ventoy.* You can now drop any ISOs you want into the USB drive and boot into it.

Another neat trick is *Ventoy's* ability to create a persistent drive inside a DAT file that you can then use with all supported live distributions, such as Ubuntu, Fedora, Mint and more. You can point the **CreatePersistentImg.sh** to your USB disk to create it. Assuming your USB disk is at **/dev/sdb**, running `sudo ./CreatePersistentImg.sh /dev/sdb` creates a 1GB **persistence.dat** file in the current directory. The script has several options to define the size of the **persistent.dat** file, along with its filesystem.

You now need to generate a **ventoy.json** file to point *Ventoy* to this persistent disk. You can either generate this file manually or use the **VentoyPlugson.sh** script like so:

```
$ sudo bash VentoyPlugson.sh /dev/sdb
```

Now head to **http://127.0.0.1:24681**. You'll get a whole list of plugins. Head to Persistence Plugin, click on Add and enter the absolute paths to the ISO file and the **persistence.dat** file.

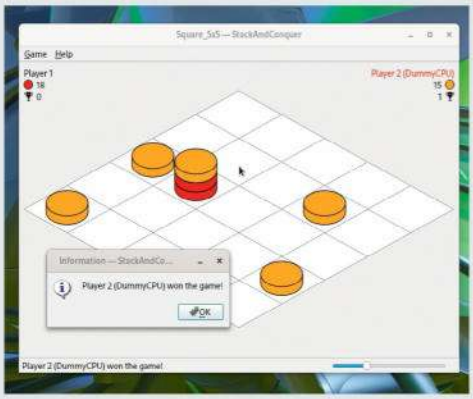


Besides the normal **GUI**, Ventoy also has a web-based interface. You can also run it in the terminal with a shell script.

BOARD GAME

StackAndConquer

Version: 0.10.1
Web: https://github.com/ElTh0r0/stackandconquer



If you can't find another human player to play against, StackAndConquer also has two AI players, a dumb one and a really clever one.

If Christopher Nolan were a game developer, he would design something like *StackAndConquer*. The objective of the game is to build a stack of pieces. The game is played between two players, with different coloured pieces, as they try to outfox each other to build a stack of pieces five units high with one of your pieces on top.

The game has an Ubuntu PPA and precompiled binaries for several distros including Fedora, Debian and OpenSUSE. However, it's best to just grab the distro-agnostic AppImage.

A player can take one of two actions: they can place one of their pieces (you have 20 in all) on an empty square, or they can move a piece or an entire stack of pieces on top of another, which is what you'll be doing for the most part.

This might seem straightforward, but there are certain rules for moving pieces.

Pieces are always taken from the top of a stack. They can only move at right angles or diagonally. Additionally, stacks can't cross other pieces or stacks, but they can be split at any level, so you can move one or multiple pieces from a stack. And you can move your opponent's pieces and even stacks with an opponent's pieces on top.

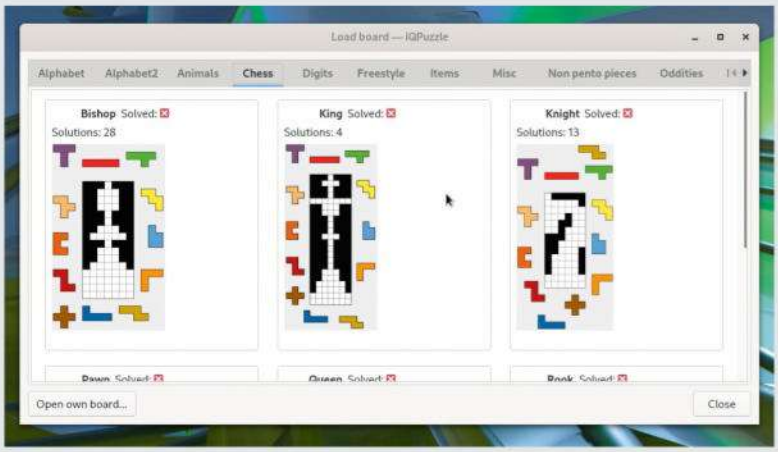
Moreover, it is the height of the target stack that determines the length of a move. So, a two-piece stack can move three places if the target stack is three pieces high. But the same three-piece stack can not move three places on top of the two piece stack. To help you, when you select a stack, the game highlights the towers it can be moved to.

When a player manages to create a stack of five or more pieces, the player whose piece is on top wins the game.

PUZZLE

iQPuzzle

Version: 1.3.1 **Web:** https://elth0r0.github.io/iqpuzzle/



The different boards in iQPuzzle are shaped to match their name. For instance, the Chess boards are stylised to look like the different chess pieces.

If *StackAndConquer* sounds too challenging, here's something that at least looks familiar. But while *iQPuzzle's* pieces might look like *Tetris*, they aren't; *iQPuzzle* is a set of pentomino puzzles, where you have to use the available pieces to fill up the canvas. Oh, and *Tetris* was inspired by pentomino puzzles, but only uses four-block pieces.

iQPuzzle is available in several formats, including the distro-agnostic AppImage and Flatpak. You can download the AppImage and give it executable permissions, either from the file manager or with `chmod +`. If you prefer Flatpaks, you can install *iQPuzzle* with `flatpak install flathub com.github.elth0r0.iqpuzzle`.

The game has over 300 puzzles divided into various categories. There's one called Alphabet with puzzles labelled A, B, C and so on. There's another called Animals that has puzzles named after creatures such as Bird, Boar, Chameleon and such.

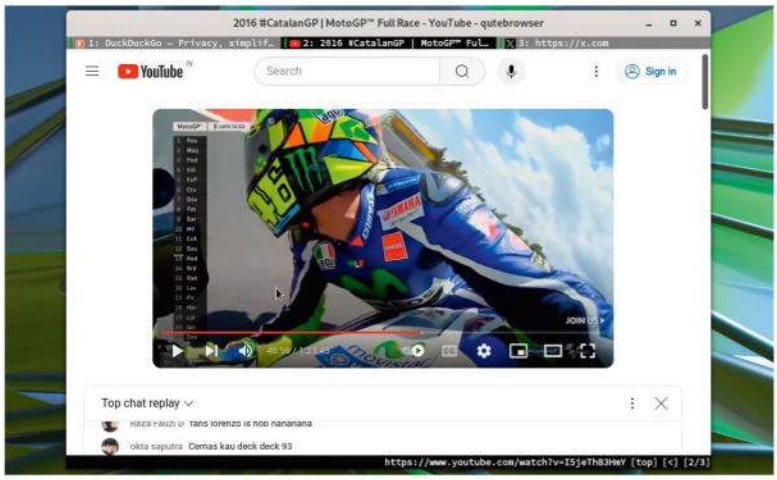
On launch, the game loads up a simple rectangle puzzle, which has about 2,339 solutions. You have to drag the pieces to fill up the board. You can just as easily drag pieces outside the board. You can right-click on a piece to flip the block, or use the scroll wheel to rotate it. To change these mouse controls, head to Settings > Configure iQPuzzle to bring up the game's preferences.

The game tracks your progress through the time it takes to solve a puzzle. Solved puzzles are marked as such. You can start playing a new puzzle by heading to Game > Choose A New Game. This brings up a tabbed list of boards. You can preview the board and the game also lists the number of possible solutions for every board. If you'd prefer to choose games by their difficulty, go to Game > Random Game, and select from Easy, Medium or Hard.

WEB BROWSER

Qutebrowser

Version: 3.2.0
Web: www.qutebrowser.org



While the mainstream web browsers are going all out to outdo each other, here's one that walks a different line. *Qutebrowser* is a minimalist browser that's got an ace up its sleeve. While Linux users have a handful of very capable minimalist browsers, *Qutebrowser* sets itself apart with its keyboard-driven approach.

The browser is available in the repos of virtually every desktop browser. Ubuntu users can fetch it with `sudo apt install qutebrowser`, while Fedora users can use `sudo dnf install qutebrowser` to install it.

On first launch, the browser opens two tabs, one with a quick-start guide on its website, and the Duck Duck Go search engine in another. The first thing you notice about *Qutebrowser* is its minimal GUI. You know the browser supports tabs because two are open right in front of you, but there are no buttons to open a new tab. In fact, there's not even an address bar.

Unsurprisingly, working with *Qutebrowser* takes some doing if you're used to the GUI-heavy mainstream browsers. But if you're a CLI stalwart familiar with *Vim*, you'll feel right at home. The quick-start guide is a good place to help you get orientated with the browser as it lists the keyboard shortcuts for the most-used functions.

To go to a new web page, press `o`, type a URL, then press Enter. If you want to open a new URL in a new tab, use `O`. The browser supports the common shortcuts you'll find in most mainstream browsers. For instance, Ctrl+t opens a new tab as well, which by default heads to the DDK search engine, and Ctrl+w closes the active tab.

Use `:` to open a long list of supported actions. As you begin typing, the browser starts narrowing down the list of actions.

It might be minimal and not look the part, but you can use Qutebrowser to watch videos, check your email, and scroll through tweets and toots.

FEED READER

RSS Guard

Version: 4.7.2 **Web:** https://github.com/martinrotter/rssguard

If you use a feed reader to get your daily dose of news, you'll like *RSS Guard*. It has a nice-looking interface and a rich set of features, and supports all of the commonly used feed formats, including RSS/RDF and ATOM.

RSS Guard is available as an AppImage that you can download from its website and make executable with `chmod +x`. It's also available on Flathub and can be installed with `flatpak install flathub io.github.martinrotter.rssguard`.

When you fire it up for the first time, *RSS Guard* asks you to add an account. The app supports online feed synchronisation with virtually all the popular feed services, including Feedly, Tiny Tiny RSS and several others. If you don't use any of these services, you can just select the RSS/RDF/ATOM/JSON option to manually add feeds.

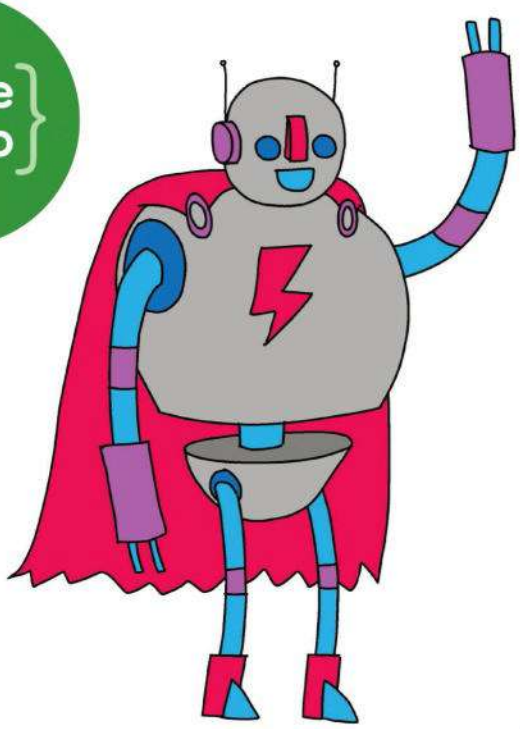
When you take that route, the app asks you to select a title and an icon for the feeds. This is handy to separate these manually added feeds from those imported from other services. Then proceed to review the other settings, though it isn't a bad idea to just go with the defaults.

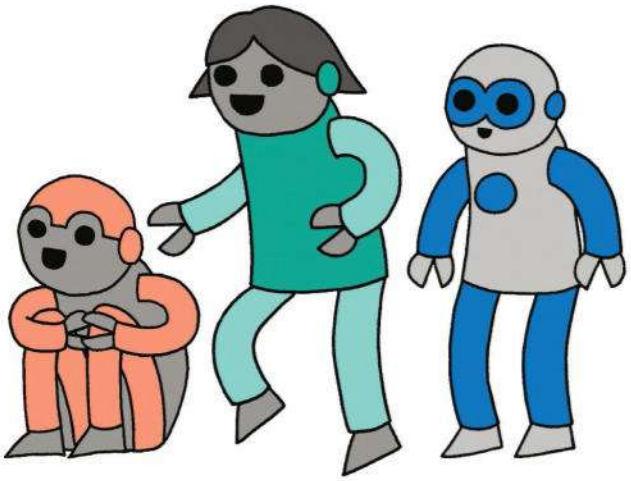
After running through the account creation process, *RSS Guard* asks whether it should add a default set of feeds. Again it's a good idea to do so, and you can always prune this list later.

To manually add feeds, head to Feed > Add Item > Add New Feed. The best thing about *RSS Guard* is that you don't need to know the exact URL of the feed. Just enter the URL of the website and click the Discover button. The app analyses the website and digs up all the feeds it hosts. Select the feed you want, choose a target parent folder, and press Import Checked Feeds. LXF

If you subscribe to a podcast, you can use RSS Guard to stream it via the app's built-in media player, which also gives you the option to download episodes.

code club





Can you help inspire the next generation of coders?

Code Club is a nationwide network of volunteer-led after school clubs for children aged 9-11.

We're always looking for people with coding skills to volunteer to run a club at their local primary school, library or community centre for an hour a week.

You can team up with colleagues, a teacher will be there to support you and we provide all the materials you'll need to help get children excited about digital making.

There are loads of ways to get involved!

So to find out more, join us at **www.codeclub.org.uk**

CLASSIC DEMOS

Create real-time water effects

Part Four! Don't miss next issue, subscribe on page 16!

Ferenc Deák scared the **LXF** team with overly abusive mathematical formulae last month. Thankfully, that's all water under the bridge now.

OUR EXPERT

Ferenc Deák is still stuck in the past for this instalment, but at least now he has cleaned his fridge.

Previously we have been experimenting with textures and mathematics, and have presented some effects that were heavily used in the compos of the old days. These effects are based on the distortion and clever manipulation of textures by algorithmically choosing texel points and presenting them on the screen using a formula.

Given the expansive scope of texture manipulation and its numerous applications, we are now introducing a complex effect based on this mechanism, which will yield surprisingly interesting results.

The effect we are presenting is the infamous water effect. Water can have several manifestations, such as raindrops, waves, a splash in the ocean – and all of these need similar but slightly different approaches. We are presenting the simplest of all: ripples caused by raindrops on a water surface. To keep things simple, we will have just one raindrop. Or maybe it's a dripping tap. When implemented properly, we'll see something like the blue screenshot *(above-right)*.

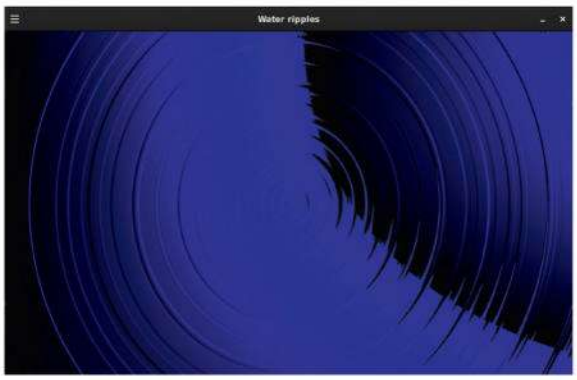
Water you thinking?

Implementing water ripples is a bit more complex than anything we have done before, due to the very dynamic nature of the effect. As you can see in the screenshot, the ripples are nothing more than concentric circles, which grow in radius from the drop point, until they reach their maximum size, and then, after a while, calm down to their zen stage of nothingness.

The real-time calculation of drop circles for the effect would not be feasible, due to the complexity of the formulae, but we can come up with a very good approximation. We also want to keep things simple, as real-time simulation of fluids is a highly complex and energy-consuming task, so we're just presenting the most basic and necessary elements to make this work.

In order to have a good approximation, we need at least two steps to the process; practically, this implies that the current state is dependent on the previous state. So, in order to draw the ripples for the current frame, we need to know how the ripples were drawn in the previous state. This can be dealt with as follows:

```
int heightMap[2][SCREENSIZE_X * SCREENSIZE_Y] = {0};
```



What do you mean, it looks like a vinyl record?

Let's not forget that we are manipulating a texture to behave like water that is being rippled, so this explains the name of the variable; it keeps the height of the various ripples at their current and previous stage for each pixel on the screen. In order to keep the sensation of movement, we continuously recalculate this map. Each step updates either the first or the second element of the array based on the 'other' stage of the map. This dynamic exchange of array indexes is done by a very simply trick in the **updateScreen** method that should already be familiar (if not, grab a previous issue of the magazine and check it out).

```
void updateScreen(Uint8* screen, const
std::vector<int>& imageData) {
  static int dropletRadius = 5, currentHeightMapIndex =
0, dropletCounter = 0;
  dropletCounter++;
  calculateWater(currentHeightMapIndex ^ 1,
WATER_WOBBLITY);
  for (int cc = 0; cc < dropletCounter; cc++) {
    waterDroplet(SCREENSIZE_X/2,SCREENSIZE_Y/2,cc
*dropletRadius,dropletRadius*10,
currentHeightMapIndex);
    smoothenWater(currentHeightMapIndex);
    dropletRadius+=4; }
  if (dropletCounter == 15) {
    dropletRadius = 0;
    dropletCounter = 0;}
```

```
  drawWater(currentHeightMapIndex, imageData,
screen);
  currentHeightMapIndex ^= 1; }
```

The **updateScreen** for the water effect starts with initialising three variables, two of which are obviously the number and size of the droplets being drawn in the current iteration. The third, **currentHeightMapIndex**, is the one used to calculate which height map we are working on.

The method, after initialising the **static** variables (as we all know, static variables are initialised only once in their lifetime), calls the **calculateWater** for the 'other' height map, which is indicated by the `currentHeightMapIndex ^ 1` expression. The ^ operator is the infamous **XOR** operator – we will present it a few lines below, so hang on tight.

WATER_WOBBLITY is a funny name, it just represents how 'wobbly' the water is – in other words, the density of the liquid into which we are dropping a droplet. Feel free to experiment with this value – you will have some interesting results. The **calculateWater** method is responsible for calculating the next stage of the water height maps, and is as follows:

```
void calculateWater(int currentPage, int density) {
  int count = SCREENSIZE_X + 1;
  int* newptr = &heightMap[currentPage][0];
  int* oldptr = &heightMap[currentPage ^ 1][0];
  int y = (SCREENSIZE_Y - 1) * SCREENSIZE_X;
  while (count < y) {
   int x = count + SCREENSIZE_X - 2;
   while(count < x) {
    int newHeight = ((heightSum(oldptr, count)) / 8) -
newptr[count];
    newptr[count] = newHeight - (newHeight / density);
    count++; }
   count += 2;}}
```

You may have recognised a distant relative of the averaging algorithm that we used in the fire routine, which adds up the values of pixels around a certain pixel and calculates the value of the current pixel based on the sum (provided by the **heightSum** method).

The height map **at a** specific **moment in time.**

In **calculateMap**, the value of the next iteration for the effect is calculated and stored in the **newptr** variable, which is just referring to the 'next' element of the **heightMap**. Then the method ends and we continue our explanation of the **updateScreen** method. The next sequence of code is the following **for** loop:

```
for (int cc = 0; cc < dropletCounter; cc++,
dropletRadius+=4) {
  waterDroplet(SCREENSIZE_X / 2, SCREENSIZE_Y / 2,
cc * dropletRadius, dropletRadius * 10,
currentHeightMapIndex);
  smoothenWater(currentHeightMapIndex);}
```

As easily guessed from the instructions, this sequence is responsible for drawing the **dropletCounter** count of water droplets with the specific radius. In order to have nice concentric circles, this droplet radius is being incremented by the value of 4 – this will give a nice distribution of concentric circles around the centre point of the screen.

QUICK TIP

The online repository contains a lot more code than can be fitted on these pages. Check out https://github.com/fritzone/lxf-demologia.

» PROCEDURAL TEXTURE GENERATION

Procedural texture generation is an alternative technique used in computer graphics to create textures algorithmically rather than manually painting or capturing them, or more recently, using an AI interface to generate the wanted texture. Instead of relying on premade images, procedural textures are generated dynamically at runtime based on mathematical functions, algorithms or noise patterns.

Common techniques include Perlin noise, fractal patterns, cellular automata, and mathematical functions such as sine waves or Perlin worms. These techniques can be combined and manipulated to create textures resembling natural phenomena such as terrain, clouds, wood grain or marble.

For our water texture, the following routine was used to generate the texture that we have used for underlying the droplets:

```
std::vector<int> palette;
std::vector<int>
imageData(SCREENSIZE_X *
SCREENSIZE_Y, 0);
for (int x = 0; x < SCREENSIZE_X; x++)
  for (int y = 0; y < SCREENSIZE_Y; y++) {
   imageData[y * SCREENSIZE_X + x] =
(int)( sin ((float)x / SCREENSIZE_X)  *
cos( (float)y / SCREENSIZE_Y) * 255); }
SDL_Color colours[256] = {0};
for (size_t i = 0; i < 256 * 4; i += 4) {
  colours[i / 4].r = colours[i / 4].g = 0;
  colours[i / 4].b = i/2;
  colours[i / 4].a = 255; }
colours[255] = {255,255,255,255};
SDL_SetPaletteColors(surface->format-
>palette, colours, 0, 255);
```

This sets each pixel (**x,y**) on the texture (represented by the **imageData** variable) to the value of **$cos(x) * sin(y)$** adjusted to fit in the **$[0,255]$** interval, and then generates a palette that starts from black up to blue, repeated twice (due to the blue byte component being assigned the integer value **i/2**, which will overflow at a certain point and restart from 0). If you don't divide, you get four repetitions, and if you divide by four, there are no repetitions.

Don't let the ugliness of the formula scare you – feel free to experiment with it as much as you like, and we'd love to see the generated textures. For now, this should do it, but please consider this to be just a superficial introduction to the dark arts of trigonometry. If you are seriously curious about this topic, please do read up on the subject on the world wide web, where there's a gazillion of good resources.

After drawing the current droplet, the method **smoothenWater** is called which, as the name suggests, is responsible for smoothing out the height map, to have a much smoother edge for the concentric circles.

The real drawing of the current droplet happens in the **waterDroplet** method, which is presented below:

```
void waterDroplet(int x, int y, int radius, int height, int page) {
 int radsquare = pow(radius, 2);
 float length = RIPPLE_HEIGHT / pow(radius, 2);
 height *= pow(RIPPLE_HEIGHT, 3);
 int left = -radius, right = radius, top = -radius, bottom = radius;
 if (x - radius < 1) left -= (x - radius - 1);
 if (y - radius < 1) top -= (y - radius - 1);
 if (x + radius > SCREENSIZE_X - 1) right -= (x + radius - SCREENSIZE_X + 1);
 if (y + radius > SCREENSIZE_Y - 1) bottom -= (y + radius - SCREENSIZE_Y + 1);
 for (int cy = top; cy < bottom; cy++) {
  for (int cx = left; cx < right; cx++) {
   int square = cy * cy + cx * cx;
   if (square < radsquare) {
    int dist = sqrt(sin(square * length) + sin(square * length));
    heightMap[page][SCREENSIZE_X * (cy + y) + cx + x] += (int)(((dist) * RIPPLE_DENSITY) * (height)) / ( pow(RIPPLE_HEIGHT, 4)); }}}}
```

The method does a few initialisations and boundary checks to make sure the droplet will be drawn in the correct boundaries of the screen. Then, in the double **for** loop, it does the calculation using a square root of sum of squares formula, which is very similar to the formula presented last month, when we were discussing the tunnel effect.

After several iterations, this leads to our graphical representation of the height map *(see previous page)*. For creating this example image, we used a simple black, grey and white palette, where the lighter the colour, the higher the value in the height map.

But now it's time to return to our **updateScreen** method and clarify the remaining lines of code. There aren't too many of them. The two **if** statements just make sure that after a sensible time we will turn off the tap and the source of the droplets will stop, thus giving the sensation that the dropping of droplets stops.

The line `currentHeightMapIndex ^= 1;` is a clever hack to flip the current height map index with the previous one, thus keeping the scene in motion. The really interesting part, however, is the **drawWater** method, which applies the water effect to the texture that is being sent to the method:

```
void drawWater(int page, const std::vector<int>& imageData, Uint8* screen) {
 int* ptr = &heightMap[page][0], i = SCREENSIZE_X + 1, y = (SCREENSIZE_Y - 1) * SCREENSIZE_X;
 while(i < y) {
  int x = i + SCREENSIZE_X - 2;
  while(offset < x) {
   int dx = ptr[i + 1] - ptr[i] ;
   int dy = ptr[i + SCREENSIZE_X] - ptr[i];
   auto idx = (i + SCREENSIZE_X * dx + dy) % (SCREENSIZE_X * SCREENSIZE_Y);
   int c = imageData[idx];
   screen[i++] = (c < 0) ? 0 : (c > 254) ? 254 + (LIGHT ? 1 : 0) : c; }
  i+=2;}}
```

Although it seems complicated, the explanation is quite simple: this method loops over all the pixels in the screen and sets the current value of it based on the texture that is being sent in as a parameter after the water effect is applied to the texture.

While doing this, it calculates a slight displacement of the height map, using the **dx** and **dy** values, and from the displacement value, generates the index of the texel in the texture (represented by the **imageData** vector) and sets the screen at the current coordinate in accordance with the displaced value from the height map. Due to the way the original palette and the texture is being generated (more details will follow – no need to worry yet), the value **255** for the colour index is reserved to a bright value, and if the **LIGHT** variable is set to **true**, it also applies a bright light to the current pixel.

So, we have concluded the **drawWater** method, and can head back to the **updateScreen**, where the only unexplained line is the following:

```
currentHeightMapIndex ^= 1;
```

» DEAR DEMOSCENE EFFECTS ENTHUSIASTS,

As we reflect on the four tutorials we've journeyed through together, exploring the intricate world of 2D demoscene effects, a wind of melancholy blows over us. For us, this was a nostalgic journey into the past, filled with exploration, excitement and lots of code. While we have delved into mesmerising visuals, captivating animations and mind-bending effects, we can't cover every effect. The world of demoscene effects is vast, overflowing with creativity and innovation, and trying to encapsulate it all in a handful of tutorials is daunting.

Yet, as one chapter closes, another beckons to be written. Despite the limitations of time and space, our commitment to unravelling the mysteries of the demoscene remains unwavering. Hence, at a certain point in the near future, we will delve into the realm of 3D effects, and create another mini series for uncovering its secrets.

In these forthcoming episodes, we'll be embarking on a journey through the depths of three-dimensional space, exploring the complexities of point clouds, polygonal landscapes, intricate lighting techniques and dynamic simulations. From mesmerising particle systems to breathtaking texture mapping, we'll unravel the secrets behind some of the most awe-inspiring demoscene creations.

While we may never cover every effect, technique or masterpiece crafted within the demoscene community, our journey is fuelled by curiosity, passion and a relentless pursuit of knowledge. So, as we eagerly anticipate the adventures ahead, let's celebrate the creativity, ingenuity and boundless possibilities that define the demoscene.

Stay tuned for the next chapter in our exploration: a journey into the captivating world of 3D demoscene effects, coming soon on the pages of everyone's favourite, *Linux Format*.

This is an old trick from back in the day, when every bit was important in order to have the smallest possible code size. It uses the ^= (**xor-equal**) operator to:

- Take the current value of **currentHeightMapIndex**.
- Perform a bit-wise **XOR** operation with **1**.
- Assign the result back to **currentHeightMapIndex**.

In simpler terms, if **currentHeightMapIndex** is currently **0**, after executing `currentHeightMapIndex ^= 1;` , its value will become **1**. If **currentHeightMapIndex** is **1**, its value will become **0**. This is because **XOR**ing **1** with **0** results in **1**, and **XOR**ing **1** with **1** results in **0**.

If we'd wanted to be more verbose, that single line of code is equivalent to the following code sequence:

```
if(currentHeightMapIndex == 1)
  currentHeightMapIndex = 0;
else
  currentHeightMapIndex = 1;
```

Ain't it more short, concise and beautiful?

Rain keeps falling on my fish...

Now we have one water droplet in place, there is nothing to stop us from implementing another. And another. And lots of droplets will end up as a little drizzle. There is nothing stopping us from adding more than one water droplet falling down on some unsuspecting teleosts.

In order to do that, we need to have a way to keep track of all the water drops that we have created, for example in a very simplistic structure, like the following:

```
struct Droplet {
  int x, y;
  int maxRippleCount;
  int radius;
  int maxRadius;
  int rippleCtr; };
```

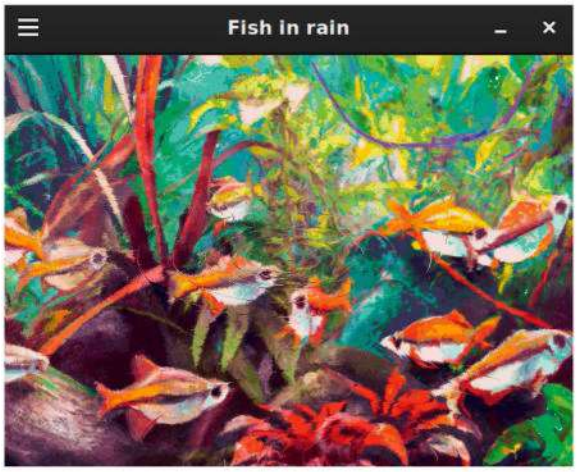
The structure contains all the necessary information for a water droplet, such as the coordinates of where the droplet fell, how many ripples the droplet has, the radius of the smallest water circle, the largest value for the outer water droplets' circle, and how many ripples the droplet has.

To keep track of all the droplets we have, we just store them handily in a large enough container:

```
std::vector<Droplet> droplets;
```

The handy container **std::vector** enables us to store a random number of droplets, without having any concerns regarding its size. In order to incorporate the new logic with several droplets, we have modified the **updateScreen** method to handle the vector of droplets in the following manner:

```
for (int i = 0; i < droplets.size(); i++) {
  droplets[i].ctr++;
  calculateWater(currentHeightMapIndex ^ 1,
WATER_WOBBLITY);
  for (int cc = 0; cc < droplets[i].ctr; cc++) {
    waterDroplet(droplets[i].x, droplets[i].y, cc *
droplets[i].radius, droplets[i].radius,
currentHeightMapIndex);
    droplets[i].radius +=2;}
  smoothenWater(currentHeightMapIndex);
  if (droplets[i].ctr >= droplets[i].rippleCount) {
   droplets[i].ctr = 0;
```



The fish happily swim in the aquarium, not conscientising the rain above them.

```
  droplets[i].x = rand() % SCREENSIZE_X;
  droplets[i].y = rand() % SCREENSIZE_Y;
  droplets[i].radius = 1;}}
```

This is very similar to the single droplet, except that it loops through the vector of droplets, and after modifying them according to the current stage, we draw them on the screen, just as we would do with a single droplet. As a side note, however, we have modified the **waterDroplet** function a bit – the following two lines were changed to use an ellipse instead of the circle for the single water droplet:

```
int radsquare = pow(radius, 2) / 6;
...
int square = (cy * cy) + (cx * cx)/6;
```

This very simple change made it possible to approximate the points of an ellipse, thus the effect looks a bit more 3D, like the previous one with the single droplet. The only thing that remains for now is to properly create all the water droplets, this is done in the **main** function:

```
srand(static_cast<unsigned int>(time(nullptr)));
int dropletCount = rand() % 15 + 15;
for(int i=0; i< dropletCount; i++) {
   droplets.push_back( {rand() % SCREENSIZE_X,
rand() % SCREENSIZE_Y / 2, rand() % 5 + 5, rand() % 25,
rand() % 15 + 5, 1});}
```

Of course, all those numbers can be changed as per your needs, so feel free to go wild with them.

The theory presented in this tutorial can be the basis of several other effects, too, such as the magical glass ball effect or the lens effect, not forgetting waves, too, because all of these effects calculate a displacement map in order to render the texture in the proper manner. However, for the moment, this particular author will take a break from the magical world of multicoloured pixels, return to the area of Linux system internals and, due to popular demand, will resume the Linux shell series of tutorials. So, if you are interested in shellology and are keen to learn more advanced aspects of the depths of Linux, it's time to dust off your old magazines (or just go online and pick up the first four instalments of the series – *see page 64 for how to order)* and do a little rereading in order to be prepared for what we will be discussing in the upcoming shell episodes. Happy coding! LXF

FYNE

Credit: https://fyne.io

Part Two! Don't miss next issue, subscribe on page 16!

Distribute your Fyne journaling app

In part two of this guide, **Andrew Williams** completes the building of a daily journal app using Fyne and prepares it for distribution.

OUR EXPERT

Andrew Williams is a software engineer and entrepreneur. He has been a core developer in *Enlightenment*, EFL and *Maven*, and founded the *Fyne* toolkit.

Last month, you met *Fyne*, a popular toolkit for building graphical applications with Go. It aims to make it easy to build native apps that work everywhere. Apps built with *Fyne* can be distributed to multiple operating systems, including Linux, BSD, Android, iOS, Mac and Windows. Last time, we stepped through the setup and Go source code required to build a simple journal app and run it on our computer.

In this part, we pick up with an important omission: storing the user data so it's not lost each time the app closes. We also look at using data binding for simpler code, and some additions to the feature set. Lastly, we explore how to package a *Fyne* app for each platform and even upload them to app stores and marketplaces.

Storing user data

Fyne comes with some standard storage APIs, and the most useful is Preferences. This is a place to store user data between app runs. Before we can use this, we need to update our app setup so it has a unique identifier (ID). The unique ID is commonly used for mobile applications or software distributed through an app store, so that two apps can't overwrite each other's data. The identifier is typically in reverse DNS notation, such as **com.example.myapp**. If you do not have a domain name for your software, it is common to use the location of the source code – for example **com.github.username.myapp**. In our code, we will replace **app.New()** with a new call that passes in the unique ID:

```
app.NewWithID("com.example.journal")
```

With that in place, we can remove the entries map of data from the code that we put together last month and replace the read and write of the data with preference calls. The instance of Preferences for your application is available from **App.Preferences()**, which returns an implementation of the fyne.Preferences

» STORAGE AND DATA BINDING

Once we've moved to *Fyne* Preferences for our storage and data binding to communicate changes, the code is even more concise than it was at the end of part one – here it is in its completeness:

```
package main

import (
    "time"

    "fyne.io/fyne/v2"
    "fyne.io/fyne/v2/app"
    "fyne.io/fyne/v2/container"
    "fyne.io/fyne/v2/data/binding"
    "fyne.io/fyne/v2/theme"
    "fyne.io/fyne/v2/widget"
)

const dateFormat = "2 Jan 2006"
func main() {
    a := app.NewWithID("com.example.journal")
    w := a.NewWindow("My Journal")
    var date time.Time
    entry := widget.NewMultiLineEntry()
    title := widget.NewLabel("Today")
    title.Alignment = fyne.TextAlignCenter

    setDate := func(d time.Time) {
        date = d
        dateStr := date.Format(dateFormat)
        title.SetText(dateStr)
        entry.Bind(binding.BindPreferenceString(dateStr, a.Preferences()))
        entry.Validator = nil
    }
    setDate(time.Now())
    prev := widget.NewButtonWithIcon("", theme.NavigateBackIcon(), func() {
        setDate(date.Add(time.Hour * -24))
    })
    next := widget.NewButtonWithIcon("", theme.NavigateNextIcon(), func() {
        setDate(date.Add(time.Hour * 24))
    })
    bar := container.NewBorder(nil, nil, prev, next, title)

    w.SetContent(container.NewBorder(bar, nil, nil, nil, entry))
    w.Resize(fyne.NewSize(200, 180))
    w.ShowAndRun()
}
```

As you can see, not a lot has changed, but the user data is now saved any time that it changes. In addition, we were able to remove the **OnChanged** callback to the **Entry**, because data binding takes care of this for us.

API. This provides getters and setters for many data types, such as String, Int and Bool, as well as lists of these types (such as a Go slice).

To use the preferences, we replace the map write (`entries[dateStr] = s` on line 23) with the new call to **SetString**, which uses the **dateStr** as the unique key for the data entry:

```
a.Preferences().SetString(dateStr, s)
```

When we load data to update the user interface, we previously accessed the data using **entries[dateStr]**. We can change this (line 32) to use the **String** method, which gets the data for a key in the preferences.

```
entry.SetText(a.Preferences().String(dateStr))
```

With these changes in place, you can now run the app – and it looks the same as before. However, when you make, changes they are written to disk automatically. So, you can quit the app and open it again, and your data is remembered.

Our journal app with mood functionality added – installed on a light mode laptop. It also includes the app icon discussed in this article as part of the metadata.

Using data binding

One of the best practices in app development as software evolves is to separate the graphical code from business logic and data handling. To support this, the *Fyne* toolkit provides a data binding package that provides a two-way data binding that most of the standard widgets support. By using this feature, an application can cause its user interface to update by simply changing the data source. In addition, with two-way binding, any input widget, such as the Entry we have used, updates the underlying data automatically.

In addition to working with basic data types, this binding can interact directly with the Preferences API that we used in the previous section. To make our code much neater, we will move to data binding for the journal entries. We start that by importing the **fyne.io/fyne/v2/data/binding** package. The next step is to replace our **SetText** call (which was added earlier in this tutorial) with a call to **Bind** instead. The source of the binding is a preferences string like before, so we use the same **dateStr** key. The replacement line (around line 32) looks like the following:

```
entry.Bind(binding.BindPreferenceString(dateStr, a.Preferences()))
entry.Validator = nil
```

As you can see, there was an additional line that sets the entry validator to **nil**. This is because adding data binding automatically validates our user data, but in this case, it is not needed. You will find this feature useful for other applications if you have more complex bindings, such as converting floating point values to strings or referencing a database.

Now that we have set up a data source for our Entry widget, we can completely remove the **OnChanged** handler –so you can remove the function we passed in (it should be found on lines 21-24). The data is automatically written into the preferences storage any time that the user updates the values. You can now run the app again and it works exactly as before – but with much less code!

Storing mood information

With the storage taken care of, now is a great time to add some more functionality to the application, this time to show how we use emoji. We will add a series of emoji buttons to track the user's mood. To do this, we use the same storage as above, but for the mood of each day, we use the storage key **dateStr + ".mood"**. The buttons we use will be positioned in a row to simulate a toggle where only one can be set at once.

To show which is chosen, we can set the **Importance** of each button. This is a semantic name meaning how it stands out from the user interface. Normally, buttons have **MediumImportance**, but if we set **HighImportance**, the button is highlighted using the theme's primary colour. Each button is expressed as a simple text button, but using emoji to generate an image inline (this works even if your Linux distro does not have colour emoji fonts installed). We call the **setMood** function on button press, the body of which is in the second boxout of this article *(page 97)*.

```
mood := container.NewHBox(
  widget.NewButton("😊", func() {
    setMood("happy")
  }),
  // more buttons go here
)
```

The mood buttons are packed into a horizontal row using the **HBox** container. This should then be placed into the **SetContent** call on our window before it is shown. We pass the container into parameter **2** of **NewBorder**, as that represents the bottom edge of the container. To position the items centrally inside the window (though still along the bottom row), we place the mood content into **container.NewCenter**, which positions it evenly in the middle of the space. Our **SetContent** line now looks like this:

```
w.SetContent(container.NewBorder(bar, container.NewCenter(mood), nil, nil, entry))
```

Assuming you copied the rest of the code from the mood-tracking boxout, you can now run the complete app. This time, you will see that mood information, along with your journal entry, is saved for each day.

Adding app metadata

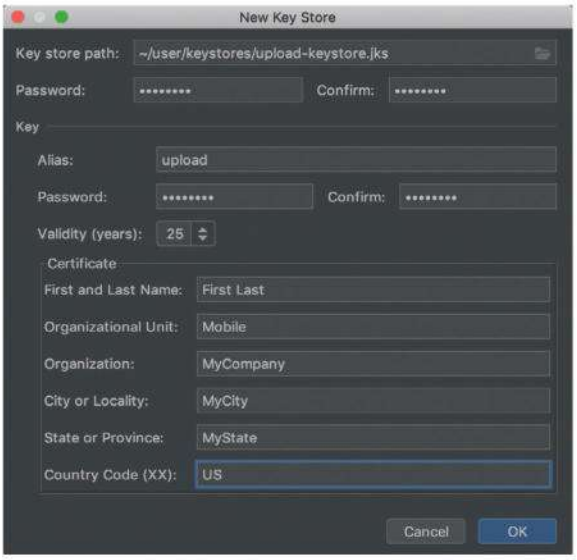
Before we can package and distribute our app, we need to gather together several items of metadata. Clearly, for app store listings, there are requirements for specific metadata and screenshots. However, even to package our app for peer distribution, we need to have the version, name and app icon prepared, along with the unique identifier that we picked earlier. These

QUICK TIP

All the code discussed in this and last month's tutorial, can be found at https://github.com/andydotxyz/linuxformat-journalapp. If you would like to explore more about how Fyne can be used visit https://fynelabs.com.

QUICK TIP

When building apps with *Fyne*, it is a good idea to avoid direct file storage or data access. The abstractions we look at in this article work identically on all devices, but mobile sandboxes and security can be problematic for lower-level code.



This is the window from Android Studio to create a new key store for uploading to Google.

items can be provided to the commands that we will execute in the packaging stage later on, but it can be easier to place them all in a single metadata file. A basic **FyneApp.toml** might look like the following:

```
[Details]
  Icon = "Icon.png"
  Name = "My Journal"
  ID = "com.example.journal"
  Version = "1.0.0"
```

In addition to creating this metadata file, we have added **Icon.png** to the project. This will be included as the icon for our app on each platform. Where a system requires the icon to be adapted in some way, this is handled automatically by the *Fyne* packaging step, which we will cover next.

Packaging to install and share

With the metadata set up and the app ready to go, we can package it for sharing or installing locally. This step is not commonly required for Go applications, as they compile to a single binary output, however a graphical app is more complex. The package format for apps is platform-dependent, so a helper tool was created to make this much easier. You can install the tool using standard Go tools:

```
$ go install fyne.io/fyne/v2/cmd/fyne@latest
```

Now that we have installed the supporting tool, it is easy to get the app installed or shared. Up to now, we have been running the code or compiled app as a command-line executable – but that means it won't be found on the app search tool or launch menu. We can fix that by installing the app to our system as a packaged piece of software. The *Fyne* tool helps us to do that with the *install* subcommand, as below:

```
$ fyne install
```

This copies the app into **/usr/local/bin** by default, along with the icon and a generated .desktop file, which describes the app. If the command was run on a different operating system, it would adapt accordingly, such as packaging into the **Applications** folder on Mac OS or installing to the Windows start menu.

The next step would be to share the app you have built with friends or colleagues, which is just as simple as a local installation. The tool has a *package* subcommand that prepares a file for sharing. In addition to the basic command, you can pass an **-os** parameter to specify a different target platform. So, to build a package for our Linux computer, an Android device and also one for a friend who uses Windows, you could do the following:

```
$ fyne package
$ fyne package -os android
$ fyne package -os windows
```

The Go programming language is excellent at cross-compiling, but the addition of the hardware-accelerated graphics that *Fyne* uses for performance makes this a little more complex. For an Android compilation, you need to have installed the NDK, which is most commonly shipped through *Android Studio* – make sure that **ANDROID_NDK_HOME** is set and *Fyne* will compile your app successfully. To target a different desktop operating system may require additional C libraries. More information about the requirements for each platform is available online at **https://docs.fyne.io/started/cross-compiling**.

For developers who don't want to manage the toolchains for each platform to release, there is an additional tool called *fyne-cross* that can manage the compilation using containers through *Docker* or *Podman*. To build the Windows package without needing to set up the dependencies, you can use *fyne-cross* to download the tools into a container, build your app, and then clean up the temporary container. If this is the approach you would prefer, try the following commands:

```
$ go install github.com/fyne-io/fyne-cross@latest
$ fyne-cross android
$ fyne-cross windows
```

Uploading to stores

As mentioned above, some of the work to prepare an application for store distribution is organising your metadata with catchy marketing text and great screenshots to go with your release – but we will assume that is taken care of. The focus of this section is how to prepare your app package for upload. We will focus on an Android release to the Google Play Store, though the steps are similar for iOS, Mac OS and other stores you may wish to ship your app to.

The first thing we need to do is to set up certification for the final binary that will be compiled. With Google Play, you can either have it generate your credentials or you can create your own. We will step through the latter as it is our preferred approach. First you need to create a Java key store for your data. This is most easily done with *Android Studio* (though an internet search will also list many other approaches). In *Android Studio,* open any project, use the Build menu and head to the Generate Signed Bundle/APK (don't worry, we won't actually use this for building). From the dialog that opens, choose Android App Bundle, then Next. You are then asked for the Key Store Path, but just tap Create New. This opens a window that enables you to set up a new key store, as shown in the screenshot *(top-left).* Fill in the information and remember your file location, alias and the password for both the store and the alias (you will need them later).

With this information, we can upload our app package to the store. Once again, we turn to the *Fyne* command-line tool that was downloaded earlier. The *release* subcommand creates a store package. We will

QUICK TIP

Each app store or marketplace has different requirements for screenshots and descriptive information. It is a good idea to create a master document and collection of high-resolution images to work with before you start uploading.

ADD MOOD-TRACKING TO OUR APP

To add more functionality to our journal app, we can also track mood with a few additional buttons in the app. The following code should be inserted into the main function above the definition of the **setDate** function:

```
var updateMoods func()
setMood := func(mood string) {
  dateStr := date.Format(dateFormat)
  a.Preferences().SetString(dateStr+".
mood", mood)
  updateMoods()
}

mood := container.NewHBox(
  widget.NewButton("😀", func() {
    setMood("happy")
  }),
  widget.NewButton("😍", func() {
    setMood("loved")
  }),
  widget.NewButton("😴", func() {
    setMood("sad")
  }),
  widget.NewButton("😫", func() {
    setMood("tired")
  }),
  widget.NewButton("😬", func() {
    setMood("stressed")
  }),
)
updateMoods = func() {
  setSelected := func(o fyne.
CanvasObject, high bool) {
    b, ok := o.(*widget.Button)
    if !ok {
      return
    }

    if high {
      b.Importance = widget.
HighImportance
    } else {
      b.Importance = widget.
MediumImportance
    }
    b.Refresh()
  }

  dateStr := date.Format(dateFormat)
  dayMood := a.Preferences().
String(dateStr + ".mood")
  setSelected(mood.Objects[0],
dayMood == "happy")
  setSelected(mood.Objects[1],
dayMood == "loved")
  setSelected(mood.Objects[2],
dayMood == "sad")
  setSelected(mood.Objects[3],
dayMood == "tired")
  setSelected(mood.Objects[4],
dayMood == "stressed")
}
```

To use the above code, we insert **updateMoods()** into the end of the **setDate** function and add **moods** to the **NewBorder** in the second parameter (bottom) when calling **SetContent** *(see Storing Mood Information, p.96, as well).*

also provide a few additional parameters – in the case of Android, it is the **keyStore** (the path from above) and the **keyName** (called alias in *Android Studio).* With that information, the command looks like this:

```
$ fyne release -os android -keyStore ~/path/to/
keyStore.jks -keyName myKey
```

Once this command completes, you will have an AAB file, which is what Google Play expects to be uploaded to the store. Before you can upload your app, you need to create a new app on the Google Play Store console and add all the metadata and screenshots you have prepared, ensuring that the unique identifier of the new app on the Google Play Store matches what you have been using earlier. You also need to upload the signature we set up. The link reads 'Upload a new app signing key from Java Keystore' but there is full documentation at **https://developer.android.com/studio/publish/app-signing#enroll_existing**.

Once complete, you can prepare the version for release, and when the website asks you to 'Upload app bundle', you can just drag the AAB file we have created into the browser window. Be sure to complete the process of agreeing to the terms and adding app compliance information. You are then ready to submit the app for approval. In around a day or so, your new application should be approved and available on the Google Play Store.

The process for iOS mobile/tablet apps is very similar. However, with Apple you need to request a provisioning profile and certificate from the company. The Apple documentation is also very helpful to explain this process: **https://developer.apple.com/help/account/manage-profiles/create-an-app-store-provisioning-profile/**. Note, however, that the licensing restrictions require you to compile the iOS app on a Mac OS computer. The *Fyne release* command works as described above, using iOS as the target operating system. In the case of iOS, the *release* command outputs an IPA file, while for Mac OS distribution, it outputs a PKG bundle. Also note that to distribute to the Apple App Store, it requires you to use its *Transporter* app, which is only available for Mac OS.

Fyne art

In this article, and the previous instalment, we have walked through building a graphical application with the *Fyne* toolkit and Go programming language. In this second part, we found that it is indeed simple to get this app built for, and distributed to, all the different platforms and devices in common usage today. Hopefully, you have found this simple to follow, thanks to the carefully curated APIs of both Go and *Fyne* – especially in comparison to the toolkits Linux desktop apps have been built with to date. If this article has sparked your interest, you can find many other apps that are open source at **https://apps.fyne.io**, so you can learn from their approach. The *Fyne* community is online across Slack, Discord and Matrix, with links to all from its website at **https://fyne.io**.

All the code discussed in this and last month's tutorial, along with the metadata and images used, can be found at Andrew's GitHub repository at **https://github.com/andydotxyz/linuxformat-journalapp**.

If you would like to explore more about how *Fyne* can be used in business, or the products and services available to make your app development more productive, please visit **https://fynelabs.com** or get in touch to discuss your next project. LXF

QUICK TIP

The Preferences API provides you with a great way for simple storage of user data. But it is also a great way to future-proof your app. If you add a cloud storage provider later (with App.SetCloud Provider) this data will be synchronised to the cloud!

NEXT MONTH

LXF319 will be on sale Tuesday 20th August 2024

IT'S OUT! LINUX MINT 22

Get started with the greatest release of Linux Mint to date: new apps, installer, backup and no Snaps!

Remote control made easy

Connect and control your remote systems with total security, full desktop access and low-bandwidth modes.

Google Matters

The next-gen smart-home device is here: Matter. We look at building and controlling our own open source versions.

Tech-dense NixOS

A Linux distro with all the hot tech you could want: immutable design, atomic updates, declarative config!

What happened to Symbian?

We relive the rise and fall of the ultimately open source Symbian OS that was crushed under Android boots.

Content of future issues subject to change. We could be still waiting for it to stop raining...

CREDIT: Magictorch

LINUX FORMAT
The #1 open source mag

Future Publishing Limited, Quay House, The Ambury, Bath, BA1 1UA
Email contact@linuxformat.com

EDITORIAL
Editor-in-chief Neil Mohr
Art editor Fraser McDermott
Production editor Katharine Davies
Group editor-in-chief Graham Barlow
Group art director Warren Brown
Editorial contributors
Denise Bertacchi, Jonni Bidwell, Neil Bothwick, Stuart Burns, Christopher Coke, Ferenc Deák, Shane Downing, Nate Drake, Michael Graw, Tam Hanna, Matthew Holder, James Holland, Phil Iwaniuk, Jon Masters, Nick Peers, Les Pounder, Michael Reed, Mayank Sharma, Shashank Sharma, Andreas Theodorou, Robin Valentine, Andy Williams, Mike Williams.
Cover illustration Magictorch.com
Ubuntu is a trademark of Canonical Limited. We are not endorsed by or affiliated with Canonical Limited or the Ubuntu project.
Raspberry Pi is a trademark of the Raspberry Pi Foundation.
Tux credit: Larry Ewing (lewing@isc.tamu.edu).
Linux Mint is copyrighted 2006 and trademarked through the Linux Mark Institute.
Content production Adequate Media Limited

ADVERTISING
Commercial sales director Clare Dove
clare.dove@futurenet.com
Advertising director Lara Jaggon
lara.jaggon@futurenet.com
Account director Andrew Tilbury
andrew.tilbury@futurenet.com

INTERNATIONAL LICENSING
Head of print licensing Rachel Shaw
Linux Format is available for licensing and syndication.
To find our more contact us at licensing@futurenet.com
or view our content at www.futurecontenthub.com

NEW SUBSCRIPTIONS & BACK ISSUES
Web www.magazinesdirect.com
UK 0330 333 1113 **World** +44 (0) 330 333 1113

EXISTING SUBSCRIPTIONS
Web www.mymagazine.co.uk
UK 0330 333 4333 **World** +44 (0) 330 333 4333
Subscription delays: Please allow up to seven days before contacting us about a late delivery to **help@magazinesdirect.com**

CIRCULATION
Newstrade & retail category director Ben Oakden

PRODUCTION AND DISTRIBUTION
Group head of production Mark Constance
Production manager Nola Cokely
Senior ad production manager Jo Crosby
Digital editions manager Jason Hudson

THE MANAGEMENT
Managing director technology group Paul Newman
Global head of design Rodney Dive
Commercial finance director Tania Brunning
Printed by William Gibbons & Sons
Distributed by Marketforce UK
121-141 Westbourne Terrace, London W2 6JR. **www.marketforce.co.uk**
For enquiries email: **mfcommunications@futurenet.com**

Order and access back issues: If you are an active subscriber, you have instant access to back issues through your iOS or Android device/s. Your digital magazine entitlement is available at no additional cost and no further action is required. Pocketmags library may not have access to the full archive of digital back issues. You will only be able to access the digital back issues as long as you are an active subscriber.

To purchase single back issues (print format only): Visit www.magazinesdirect.com (click on Single Issues tab) or email help@magazinesdirect.com. For further help call +44(0)330 333 1113. Lines are open Mon- Fri 8.30am-7pm and Sat 10am-3pm UK time. Magazinesdirect.com is owned and operated by Future Publishing Limited.

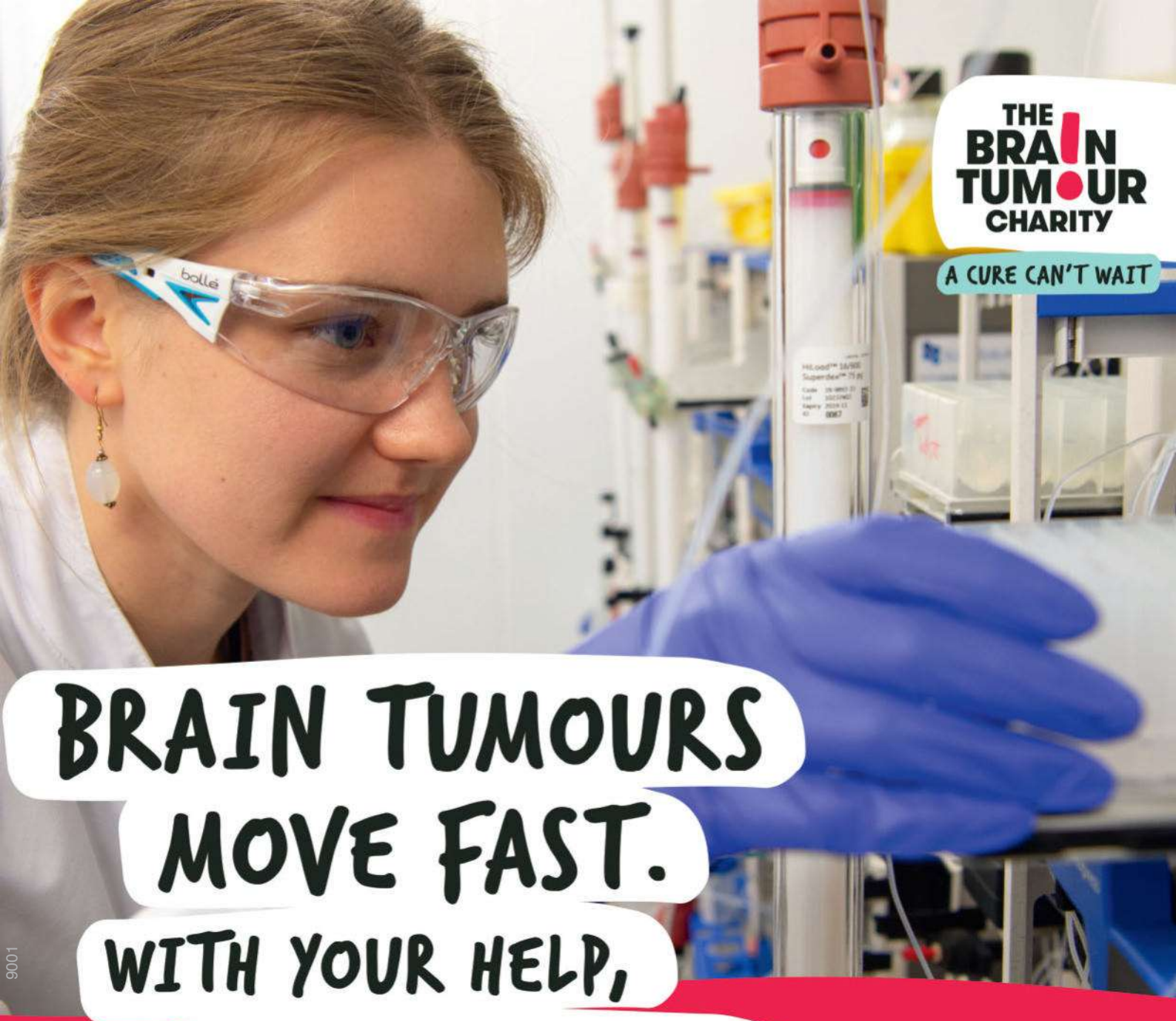
BAR rates for Linux Format: £84.37 for UK, €171 for Europe, $194 for USA, £149 for rest of world.

The Electronic Frontier Foundation is the leading nonprofit organization defending civil liberties in the digital world. Founded in 1990, EFF champions user privacy, free expression, and innovation through impact litigation, policy analysis, grassroots activism, and technology development. We work to ensure that rights and freedoms are enhanced and protected as our use of technology grows.

EFF.ORG

ELECTRONIC FRONTIER FOUNDATION

Protecting Rights and Promoting Freedom on the Electronic Frontier



THE BRAIN TUMOUR CHARITY

A CURE CAN'T WAIT

BRAIN TUMOURS MOVE FAST.

WITH YOUR HELP, WE CAN TOO!

We're working to create a future where brain tumours are curable.
We urgently need your help to accelerate research.

Text DEFEAT5 to 70507 to donate £5, please help us to find a cure.

thebraintumourcharity.org

Registered with FUNDRAISING REGULATOR

9001

9000